DOCUMENT RESUME

ED 112 915 IR 002 626

TITLE	Proceedings: Conference on Computers in Chemical Education and Research, Dekalb, Illinois, 19-23 July 1971.
SPONS AGENCY	National Science Foundation, Washington, D.C.
PUB DATE	Jul 71
NOTE	612p.
EDRS PRICE	MF-$1.08 HC-$31.10 Plus Postage
DESCRIPTORS	*Chemical Analysis; *Chemical Reactions; Chemistry; *Chemistry Instruction; *Computer Assisted Instruction; *Computer Based Laboratories; Computer Graphics; Computer Programs; Conference Reports; Curriculum Development; Data Bases; Information Retrieval; Information Systems; Networks; On Line Systems; Programing Languages; State of the Art Reviews
IDENTIFIERS	APL; A Programing Language

ABSTRACT

Computers have effected a comprehensive transformation of chemistry. Computers have greatly enhanced the chemist's ability to do model building, simulations, data refinement and reduction, analysis of data in terms of models, on-line data logging, automated control of experiments, quantum chemistry and statistical and mechanical calculations, and information storage including retrieval and the capacity for dynamic visual display. A recent conference addressed itself to the following aims: (1) provide a national forum for the impact of computers on chemistry; (2) encourage more investigation on computers; (3) make known the present status and the potential of computer-assisted instruction (CAI) in chemistry; and (4) inspire further examination of the chemistry curriculum in light of recent advances in computer technology. The conference report is a collection of 80 papers divided into the following categories: data fitting; curve fitting and band analysis of spectra; structure determination; CAI; computer technology in theoretical chemistry; chemistry curriculum development; communications in the field of chemistry; computer control of experiments; the use of A Programing Language (APL) in the undergraduate curriculum; and information systems. (EMH)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED112915

NSF GJ-28335

SCOPE OF INTEREST NOTICE
The ERIC Facility has assigned this document for processing to: IR

In our judgement, this document is also of interest to the clearinghouses noted to the right. Indexing should reflect their special points of view. SE

PROCEEDINGS

Conference On Computers In Chemical Education And Research

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Sponsored By The

American Chemical Society, Division of
Chemical Education

National Research Council, Division of
Chemistry and Chemical Technology

National Science Foundation

Northern Illinois University

2

Dekalb, Illinois
19 - 23 July 1971

IR 002 626

CONFERENCE ON COMPUTERS IN CHEMICAL EDUCATION AND RESEARCH

***

Conference Co-Directors:

F. M. Miller, Chairman, Department of Chemistry, Northern Illinois University.

W. Roy Mason, Department of Chemistry, Northern Illinois University.

Peter G. Lykos, Department of Chemistry; Director, Computation Center Illinois Institute of Technology; Chairman, Committee on Computers in Chemistry, National Research Council--National Academy of Sciences.

Program Steering Committee:

Peter G. Lykos, Illinois Institute of Technology, Chairman.
William Gwinn, University of California, Berkeley.
David Harris, University of California, Santa Barbara.
Joseph Lagowski, University of Texas, Austin.
John Overend, University of Minnesota, Minneapolis.
Harold Scheraga, Cornell University, Ithaca, New York.
Donald Secrest, University of Illinois, Urbana.

Session and Workshop Chairmen:

Roger Anderson, Lawrence Radiation Laboratory.
Klaus Biemann, Massachusetts Institute of Technology.
William Carnall, Argonne National Laboratory.
Joseph Denk, North Carolina Educational Computing Services.
Raymond Dessy, Virginia Polytechnic Institute.
Walter Hamilton, Brookhaven National Laboratory.
David Harris, University of California, Santa Barbara.
Jeffrey Johnson, University of Pittsburgh.
Harvey Kriloff, University of Illinois, Chicago Circle.
Joseph Lagowski, University of Texas, Austin.
David Larsen, Virginia Polytechnic Institute.
Lawrence Snyder, Bell Laboratories.
Martha Williams, IIT Research Institute.

Arrangements:

Northern Illinois University

Kenneth Kolwyck	Jeffrey Fey
Daniel Wiczer	Eleanor Stahl

Acknowledgments:

The Conference is supported in part by the National Science Foundation Grant GJ-28335

The assistance of the staff of the NIU Division of Computer Services under the direction of Gerald Duffy is gratefully achknowledged.

A STATEMENT OF PURPOSE

Electronic digital and analog computers have effected a comprehensive transformation of chemistry. They have greatly enhanced model building or simulation, data refinement and reduction, analysis of data in terms of models, on-line data logging and control of experiments, quantum chemistry and statistical mechanical calculations, and information storage, handling and retrieval including dynamic visual display.

> Sophisticated technique for data refinement and reduction, such as time-averaging, curve fitting and Fourier transforms now can be systematically applied, thus enabling the experimenter to extract much more useful information from his data.
>
> With a much larger variety of numerical techniques available to him, the chemist is no longer limited to the formulation of only those mathematical models for which analytical solutions in closed form can be found. Models can be made much more comprehensive.
>
> The rapid growth of the mini-computer industry has provided relatively inexpensive and powerful small computers, and integrated circuit module technology enables the chemist to design and construct his own interfaces between the small control computer and his experiment.
>
> Other non-numeric applications of the computer include techniques for the systematic search of the chemical literature, structure determination and molecule identification by data matching, and design of complex organic synthesis.

A major problem facing the chemical profession is that most chemists have not yet become reasonably aware of how important the computer has become in chemistry.

The problem is further compounded in that chemistry curricula have not yet been significantly content modified in order that these new techniques can become an integral part of the education of the chemistry student.

The problem is in part one of communication between those few developing these new techniques of chemical problem solving and those many who can use them and/or teach them.

Consequently, this Conference has been organized to:

- provide a national forum on the impact of computers on the way chemists do chemistry.
- encourage wider investigation of the use of computers in chemical research.
- make known the current status and the potential of computer assisted instruction, and
- inspire further examination of chemistry curriculum content in reaction to computer utilization.

TABLE OF CONTENTS

Titles and Authors	Page
"Data Acquisition and Refinement: Introduction," by Klaus Biemann	1-1
"On-Line Data Refinement," by Roger Anderson	1-2
"Computer Assisted Valuation of Error," by Ralph D. Nelson, Jr., Mark R. Ellenberger, and F. Richard Ellenberger	1-21
"Corrected Fluorescence Spectra by Computer Postprocessing," by V. Alan Mode, David H. Sisson, and Carolyn A. Bailey	1-30
"Least Squares and Statistics in Chemistry," by John Overend	1-38
"Off-Line Automation System for a Raman Spectrometer," by Bernard J. Bulkin and Arthur Noguerola	1-47
"Computer Collection of Mass Spectrometric Ionization Efficiency Curves," by Bruce R. Conard, Peter M. Fast, David L. Nordlund, and Paul W. Gilles	1-56
"Computer Assignment of Complicated Spectra: Introduction," by William Carnall	2-1
"Computer Simulation of NMR Spectra in Liquid Crystals," By Lawrence Snyder	2-5
"An ESR Lineshape Analysis for Determination of Unresolved Metal Hyperfine Splittings in Ion Pairs: Its Application to the Benzene Anion Radicals," by M.T. Jones, M. Komarynsky, and R.D. Rataiczak	2-14
"Evaluation of Complex ESR Spectra by Computer," by W.E. Bennett and R.E. Buckles	2-31
"Structure Determination of Organic Molecules by Mass Spectrometry," by Klaus Biemann	2-36
"Resolution of Electronic Absorption Spectra with Lognormal Curves," by D.B. Siano, J.A. Thomson, and D.E. Metzler	2-46
"Real-time Analysis of Ultraviolet Absorbance Spectra of one to five DNA Samples, Using CLASS, a Spectroscopy-Oriented Language on an 8K Mini-Computer," by D.C. Harris	2-58

"Structure Determination and Chemical Kinetics: Introduction," by Walter Hamilton - 3-1

"The Use of Computers in Crystallography," by James Ibers - - - - - 3-3

"A Model Study of Ketones with Rotating Groups," by S. MacKenzie, S.F. Marsocci, and Mary E. Scanlan - - - - - 3-7

"Data Acquisition and Equipment Control in Microwave Spectroscopy Using a Dedicated Digital Computer," by M.D. Harmony, D.K. Hendricksen, P.M. Fast, and D.L. Nordlund - - - - - - - - - 3-17

"An On-Line, Time Shared Computer System for Molecular Structure and Dynamics," by Kent Wilson - - - - - - - - - - - - 3-26

"Application of the Monte Carlo Method to First-Order Rate Equations," by John J. Manock and David L. Hooper - - - - - - 3-35

"Series Integration of Consecutive Rate Equations," by Norman C. Peterson and Howard J. Butcher - - - - - - - - - - 3-39

"Computer Assisted Instruction: Introduction," by Joseph Lagowski - 4-1

"Flexible Computer-Based Techniques in Chemistry Instruction," by S.J. Castleberry - - - - - - - - - - - - - - - - 4-2

"The Use of Computers in Undergraduate Laboratory Instruction," by S.P. Perone - 4-15

"Instruction in Chemistry Using PLATO," by Stanley Smith - - - - - 4-39

"Computed Assisted Instruction at Pitt," by E.E. Tratras and K.J. Johnson - 4-47

"Direct Classroom Use of a Time-Sharing Computer in Chemical Education," by Philip E. Stevenson - - - - - - - - - - - - - - - 4-54

"Theoretical Chemistry: Introduction," by Lawrence C. Snyder - - - 5-1

"Computers in Chemical Education and Research, by Arnold Wahl - - - 5-2

"MO Theory as a Chemical Tool, by Michael Dewar - - - - - - - - - - 5-18

"Computer Assisted Investigation of Liquid Water," by Frank Stillinger - 5-32

"The Salsburg Polytope Bound Algorithm: Application to the Imperfect Anharmonic Solid," by Grant Miller and Russel Larsen - - - - 5-35

"Symmetry," by Ronald Wohl - - - - 5-62

"Chemistry Curriculum Development," by Joseph Denk - - - - 6-1

"Curriculum Development Through Computer-Based Materials: Graphics, Tutorials, and Programming Languages," by Jeffrey Johnson - - - - 6-5

"Application of 'Canned' Programs, and Computerized Literature Searching to an Undergraduate Chemical Curriculum," by Larry R. Sherman - - - - 6-13

"A Computer-Based Simulation Game for Teaching Experimental Strategies in Chemical Kinetics," by H. Warren Smith - - - - 6-22

"The Vector Space Theory of Matter," by F. Al Matsen - - - - 6-30

"Introduction to Computers and Computing in the Physical Chemistry Laboratory," by M.R. James and W.G. Koch - - - - 6-48

"The Use of a Teaching Computer in the Undergraduate Laboratory," by C.E. Klopfenstein and C. Wilkins - - - - 6-54

"Chemistry, Computers, and Graphic Interaction," by Harvey Kriloff - - - - 6-60

"Communications, Networks and Program Exchanges: Introduction," by David O. Harris - - - - 7-1

"National Cooperation in Theoretical Chemistry" Quantum Chemistry Program Exchange and Possibilities for a National Laboratory," by Harrison Shull - - - - 7-2

"The ARPA Network," by Lawrence G. Roberts - - - - 7-5

"Regional Computer Networks and Curriculum Development," by Peter G. Lykos - - - - 7-13

"A Computational Laboratory for Higher Education," by Robert Dewar - - - - 7-22

"The Eastern Michigan University Center for the Exchange of Chemistry Computer Programs, EMU-CECCP," by Ronald W. Collins - 7-33

"Computer Control of Experiments: Introduction," by Roger Anderson - 8-1

"Time, Control and the Minicomputer," by Raymond Dessy - - - - - - 8-4

"Computer Control of a High-Temperature Quadrupole Mass Spectrometer," by Peter Fast, Jeffrey R. Castles, and Paul W. Gilles, - 8-19

"An Interactive On-Line System for Computer-Assisted Ion Cyclotron Resonance," by Charles L. Wilkins, Michael L. Gross, and Robert S. Marianelli - - - - - - - - - - - - - - - - 8-29

"The In-House Development and Construction of Systems for Data Acquisition and Control of Experimental Process Reactors," by C.D. Prater, W.E. Kaufman, and C.A. Williams - - - - - - - 8-38

"The Stony Brook Multi-Experiment Control System Using a Time-Shared Computer," by Yoshi Okaya - - - - - - - - - - - - 8-39

"A Computer System for Laboratory Automation," by C.E. Klopfenstein - 8-50

"The Use of APL in the Undergraduate Chemistry Curriculum," by T.R. Dehner and B.E. Norcross - - - - - - - - - - - - - - - - 9-1

"Computer-Generated Repeatable Tests in Chemistry," by F. Prosser and J.W. Moore - - - - - - - - - - - - - - - - - - 9-26

"The Effect of Computers on the Teaching of Physical Chemistry," by Morris Bader - 9-40

"A Hybrid Approach to Computer Assisted Instruction," by F.D. Tabbutt - 9-44

"Laboratory Simulation and CAI: A Rationale," by M.T. Muller - - - 9-64

"Uses of Computers in Compensatory Education," by Ralph M. Deal - - 9-79

"Armchair Unknowns," by C.G. Venier and M.G. Reinecke - - - - - - - 9-85

"Chemistry and CAI at the University of Texas at Austin: A Coordinated Approach," by G.H. Culp - - - - - - - - - - - - - 9-91

"Modern Methods of Handling Chemical Information: Introduction," - by Martha Williams - 10-1

"Modern Methods for Searching the Chemical Literature," by B. E. Holm - 10-3

"Chemical Notation a One Line Description of Chemical Structure," by C.M. Bowman - 10-14

"Machine Readable Files from Chemical Abstracts Service," by R. O'Dette - 10-32

"Handling of Varied Data Bases in an Information Center Environment," by M.E. Williams - - - - - - - - - - - - - - - - - - 10-35

"A New Approach to Computer Assisted Design of Organic Syntheses," by W.T. Wipke - 10-60

"Computer-Based Searching of the Current Chemical Literature by the Individual Scientist," by E. Kiehlmann - - - - - - - - - 10-70

"Use of the ASTM IR File for Education," by J.R. Denk - - - - - - - 10-76

Data Acquisition and Refinement

Introduction

K. Biemann
Department of Chemistry
Massachusetts Institute of Technology
Cambridge, Ma. 02139

The ever increasing availability of computers to the experimental and theoretical chemist and his increasing willingness to use them in an efficient and direct mode of interaction opens avenues not previously accessible. Detailed statistical treatment of experimental data allows much more information to be derived from them and permits the design of experiments in a way that maximizes the results while minimizing the number of data to be taken. Sophisticated error analysis further increases the confidence in the results. While the mathematical principle of the methods were largely known for a long time, only the availability of computers make it possible to apply them now on a large scale; furthermore, the arithmetic limitations are no longer such that approaches involving a seemingly vast amount of data handling would be impractical.

The advent of the small laboratory computer brought the chemist even closer to the real-time utilization of these devices and instruments directly coupled to a computer are becoming more and more widespread. The resulting ease in the accumulation of statistically meaningful amounts of data within a short time, smoothing of the data to increase the signal-to-noise ratio of the instrument and even feed-back control of the parameters of the instrument by the computer are becoming more and more widely used. Efforts of this type will undoubtedly increase the knowledge concerning chemical phenomena. Fortunately, computer techniques for storing, retrieving and correlating the resulting information will keep us from being buried under it.

TECHNIQUES IN ON-LINE DATA HANDLING*

Roger E. Anderson
Lawrence Radiation Laboratory, University of California
Livermore, California 94550

June 9, 1971

Current laboratory practice includes the use of small computers directly on-line to gather and process experimental data. More sophisticated utilization will include the use of the computer to effect or control the experiment. Simple control is referred to as open-loop, while if the processed data is used to control the experiment, it is referred to as closed-loop. This paper will be concerned with the techniques involved in obtaining the data and the techniques of on-line data processing that relate particularly to the on-line applications. Most of the material herein is not original, but is a collected set of information pertinent to on-line problems.

The first item to be considered in an on-line application is the input data. Usually, this is an analog signal that must be sampled and converted to digital form. Shannon's sampling theorem tells us that we must sample at least twice per period of the highest frequency component of information to be able to reconstruct the input signal. If signals of greater than twice the sampling frequency exist and are sampled, the high frequency sampled data will be indistinguishable from lower frequency data. This is illustrated in Fig. 1. This lack of distinguishability between two sampled frequencies is called aliasing. It must be noted that no correction of the sampled data can be made once aliasing has occurred. The manner in which the frequencies are aliased is also shown in Fig. 1 and is a periodic folding of the frequency spectrum between the Nyquist frequency and zero. The Nyquist frequency is the highest frequency represented at the given sampling rate. Figure 2[a] is an example of chromatographic peaks digitized in three methods, each having a different input data bandwidth. The bandwidths are respectively 1KHZ, 10HZ and 1HZ. The sampling rate was 20 samples per second. The aliased noise due to extreme bandwidth is obvious on the first example, and 60HZ line frequency aliased to near zero is quite apparent on the second. The third represents the true quality of the data shown in its own bandwidth.

---

*Work performed under the auspices of the U. S. Atomic Energy Commission.

The previous examples showed the results of digital data without concern for the method of analog-to-digital conversion. The previous conclusions are independent of conversion; however, some problems are conversion method dependent. Most current analog-to-digital (A/D) conversion is performed by one of two methods. These are successive approximation and integrating converters. Successive approximation converters function by successively comparing the input to a digital-to-analog signal and setting a bit at a time, starting with the most significant bit. Depending upon whether the unknown is greater or less than the comparison, the corresponding bit is retained or reset. This is shown diagramatically in the top of Fig. 3. The process is sequenced by an internal clock the total time to make a complete comparison is known as the conversion time. The A/D converter is observing the unknown voltage during the entire conversion time. We now consider what happens if the input changes during the conversion process. See the bottom of Fig. 3. If the signal should drift past a conversion level often the comparison is made, the succeeding comparisons will all be in error. However, if the change is less than the voltage corresponding to one least significant bit and linear, the result will be in error by no more than one bit. Thus the amplitude of the signal at one quarter of the maximum conversion frequency must be less than the one least significant bit. This can be accomplished by limiting the bandwidth of the signal prior to the A/D converter. This is in addition to that required to limit the bandwidth to less than the Nyquist frequency.

---

[a] From UCRL-71725, "An Automated, High Sensitivity, Three Column Gas Chromatograph," Frazer, J. W., Duval, V. and Anderson, R. E.

An alternative method to filtering for keeping the A/D converter input constant is a sample-and-hold circuit. This is essentially an electronic switch followed by an analog memory. The operation is shown diagramatically in Fig. 4. Note that it takes a finite amount of time to sample the system and achieve the input value. This is called the aperture time of the sample-and-hold amplifier. Since it takes energy from the input, and this energy is stored via a capacitor, the circuit has a response approximating an integrator. Thus we should expect a response that falls off linearly with frequency with a turnover corresponding to the aperture time. In other words, the maximum bandwidth seen by a sample-and-hold circuit is approximately one divided by the aperture time, and totally independent of the sampling rate.

The amplitude response as a function of frequency of a simple RC integrator is shown in a semi-log plot in Fig. 5. Note that this is asymptotic to two straight lines with a corner frequency equal to the time constant of the RC integrator. A gated integrator has a similar response, except that it has zeroes corresponding to the frequencies when integer multiples of the signal period coincide with the gating period. Thus, the sample-and-hold circuit gives bandwidth reduction, however the corner frequency may be very high for short aperture times.

The second most popular A/D conversion technique is the integrating voltmeter. They function in a number of methods, but all rely upon integration of the unknown signal for a period of time. Presently, the dual-slope integrator is the least expensive and its conversion method is representative. It functions by analog integration of the unknown voltage, V., for a fixed time, from t_1 to t_2. This is line Ⓐ in Fig. 6. Then a negative reference voltage is integrated until it reaches zero, from t_2 until t_3. This is line Ⓑ in Fig. 6. This time is digitally counted, and is a direct measure of the integrator voltage V_2, which is proportional to the input voltage V_1. Note that the sample integration time is less than the conversion time. This is also true for a voltage-to-frequency type of integrating voltmeter operating at less than maximum sampling rate. This has the same amplitude response as the previous example of a gated integrator. The amplitude response has zeroes and falls off reciprocally with frequency, within a corner frequency depending upon the input gate time. Thus the corner frequency increases as the duty cycle decreases, independent of the sampling rate. This is shown in

Fig. 7. And consequently, the use of an integrating analog-to-digital converter does not insure a lack of frequency aliasing. In fact, as performance (frequency-wise) improves, the aliasing becomes worse.

One more fact about analog-to-digital conversion should be pointed out. In an A/D converter, the decision is always made by an analog discriminator, a logical element. A bit or a result is made upon a greater-than or less-than comparison. This result holds until the input becomes greater than the next comparison level, although the input changes continuously between the two comparison levels. Thus one least significant bit is the minimum comparison level and any intermediate level is not rounded, but chopped to the lower comparison level. This error introduced is known as the quantization error. For true rounding, the input signal must have a voltage corresponding to one-half the least significant bit added to it prior to conversion, if the input is positive, and subtracted if the input is negative.

Now we turn to techniques by which we try to improve our data after it has been digitized and stored within the computer. The first techniques can be applied to the data on a continual basis as it is being acquired while the latter must be applied to the entire data set as a whole.

One of the first digital techniques to be used to improve the signal-to-noise ratio (S/N) of digital data was that of ensemble averaging. Ensemble averaging consists of multiple scanning of the observation in some variable domain, and summing the experimental observations corresponding to some value in the variable domain. See Fig. 7. The intensity of some measurable is determined as a function of some parameter (usually a function of time). Then the experiment is repeated and the value of the measurable corresponding to the same value of the parameter is summed. This is performed a prescribed number of times. Since the desired variable is summed in phase coherence with respect to the parameter, its signal should increase linearly with the number of scans. Noise, on the other hand, is incoherent and should increase as the root mean square of the number of scans. Thus the signal to noise should increase as the square root of the number of scans. The theory has been analyzed in detail by Ernst[b] and indicates the ultimate improvement available.

In addition to the improvement in signal-to-noise ratio, there is also an improvement in resolution. In other words, the final data has a resolution greater than that of the A/D converter used to take the data. This is shown diagramatically in the upper part of Fig. 8. When the input has a given comparison value, the digital output is that of the comparison value. If the input is less than that of the comparison value, the output is one least significant bit less. For intermediate values, the output is that of the next lower comparison value. This merely illustrates the quantization error mentioned previously. Now if the input contains random noise in addition to the signal, the input corresponds to the lower part of Fig. 8. If the noise is white noise, i.e., its frequency distribution is uniform or its amplitude distribution is Gaussian, then it has equal probability of being above or below the comparison value. When a number of measurements are taken, the average will be the mean of the measurements and will be more representative of the true value. If the noise is truly random, then its standard deviation will be proportional to the square root of the number of measurements, and since the absolute error is proportional to the standard deviation divided by the number of measurements, σ/n, the resolution will improve with the square root of the number of scans. The proportionality constant will depend upon the standard deviation of the noise signal.

---

[b] Ernst, R. R., Rev. Sci Insts. 36, 1689-95 (1965).

It is significant to point out that for data with extremely good S/N ratio, the improvement in resolution will not exist. A perfect coherent ramp will ensemble average as a staircase. However, the data can be improved by the addition of a pure noise signal at the input to the A/D converter. The data will approach linearity if the standard deviation of the noise is large compared to the quantization error. Particularly note that ensemble average does not eliminate the quantization error in any way. The data approaches the true value if the added noise has a mean of one half the least significant bit of the A/D converter. In all cases of ensemble averaging, the data taking and the experiment must be synchronized, either by computer control or by a trigger signal from the experiment.

For unsynchronized data containing a periodic component, a different digital technique is available. This is the process of correlation. Since the periodic component is coherent with its own period, the correlegram will show a periodicity with the same period. The noise in the signal is coherent only in the region of zero phase shift, and thus will contribute to the correlegram only near the origin. Thus the signal value for large values of signal difference will show the same signal to noise improvement as occurs in ensemble averaging. Since this periodicity can be obtained by source modulation, this technique gives S/N enhancement for single value measurements.

One additional on-line technique exists for the improvement of data. The technique is called digital filtering. Although long known, it was first introduced to the chemical literature by Sanitsky and Goulay.[c] The technique they described has been known by many names: digital filtering, convolution, least squares fitting, smoothing, and more. The procedure is best called digital filtering since it is directly analogous to analog filtering of the analog data prior to digitization. We can implement, in the digital domain, the same result as an analog filter in analog world. We can have low pass, high pass, band pass and notch filters. In addition, we can also implement filters for which no counterpart exists in the analog world. As a quick example, for non-real-time systems, we can implement filters with no frequency-dependent phase-shift. However, let me repeat that no digital filter can compensate for frequency aliasing errors. Once frequencies are aliased, they are digitally indistinguishable from their low-frequency counterparts. Thus the frequency domain of the digital filter lies below the Nyquist frequency. However, this does not deter its usefulness in the general case, as for example in line frequency rejection or in transmission line frequency compensation. The corresponding digital filter may not be extremely difficult to implement, while its analog counterpart can easily be almost impossible for a complicated frequency-phase-amplitude characteristic.

---

[c] Sanitsky, A. and Goulay, M. J. E., Anal. Chem. 36, 1627-39 (1964).

Digital filters are easily implemented as a sum-product of data. The data may be either direct input points, or computed input points. This divides the filters into two classes—finite memory filters which compute on input data only, and infinite memory filters that compute upon previously computed filter points. The difference between the two is easily illustrated in Fig. 9 that shows the corresponding responses to a step function of the two two-point filters. Since these filters all correspond to a moving window, they all represent a contribution. Digital filters are most easily described in terms of their Z-transform. Since that is beyond the scope of this paper, the reader is referred to the literature.[d] However, it is sufficient to say that the filter consists of a linear equation relating the output to a sum of products of coefficients times previous input or output points. Thus the filter is a finite difference equation and is directly analogous to differential equations in the analog world. As an example, the simplest five point smoothing filter is:

$$Xn = \frac{1}{5} yn\text{-}1 + \frac{1}{5} yn\text{-}2 + \frac{1}{5} yn\text{-}3 + \frac{1}{5} yn\text{-}4 + \frac{1}{5} yn\text{-}5$$

The corresponding least square filter is:

$$Xn = \frac{-3}{35} yn\text{-}1 + \frac{12}{35} yn\text{-}2 + \frac{17}{35} yn\text{-}3 + \frac{12}{35} yn\text{-}4 \quad \frac{-3}{35} yn\text{-}5$$

---

[d] Jury, E. I., "Theory and Application of the Z-Transform Method," Wiley (1964).

Although it cannot be implemented in the on-line case, the numerical Fourier transform is another example of a numerical filter. Once data is transformed from the time to frequency domain, any phase-frequency operation is a simple vector dot-product operation. The amplitude or phase at any frequency can be readily multiplied by any constant or variable to produce the desired result. However, the reader is cautioned against arbitrary constants unless he is familiar with the consequences in Fourier Series. In particular, the problems with truncation and the Gibbs Effect may produce results other than desired. Other types of numerical transforms can be employed to further treat data, but one should be very familiar with the transform properties prior to usage. A known transform does not necessarily produce desired results.

In conclusion, numerical treatment of on-line data is severely restricted by the user's knowledge of numerical properties involved. Some are very obscure indeed, and very difficult to anticipate. Thus, it is advised that the user first investigate known functions, followed by function similar to his data set prior to experimental investigation. To summarize, I quote R. Hamming[e] "The purpose of computing is insight, not numbers."

---

[e] Hamming, Richard, "Numerical Methods for Scientists and Engineers," McGraw-Hill, 1962.

BIBLIOGRAPHY

Blackman, R. B. and Tukeg, J. W., "The Measurement of Power Spectra," Dover, 1958.

Franks, L. E., "Signal Theory," Prentice-Hall, 1969.

Gold, Bernard and Rader, Charles M., "Digital Processing of Signals," McGraw-Hill, 1966.

Hamming, R. W., "Numerical Methods for Scientists and Engineers," McGraw-Hill, 1962.

Jenkins, G. W. and Watts, D. G., "Spectral Analysis and Its Applications," Holden-Day, 1968.

Korn, Granino A., "Random-Process Simulation and Measurements," McGraw-Hill, 1966.

Monroe, Alfred J., "Digital Processes for Sampled Data Systems," Wiley, 1962.

Morrison, Norman, "Introduction to Sequential Smoothing and Prediction," McGraw-Hill, 1969.

Ralston, Anthony, "A First Cause in Numerical Analysis," McGraw-Hill, 1965.

FIGURE CAPTIONS

Figure 1. Sampled Aliasing
Figure 2a. Bandwidth Chromatogram
Figure 2b. $10H_z$ Bandwidth Chromatogram
Figure 2c. $1H_z$ Bandwidth Chromatogram
Figure 3a. Successive Approximation A/D Converter
Figure 3b. Successive Approximation Converter With Drift
Figure 4. Sample and Hold Operation
Figure 5. Amplitude Response of an RC Filter
Figure 6. Dual Slope Integrating Analog-To-Digital Converter
Figure 7. Amplitude Response of Gated Integrator
Figure 8a. Quantization Error
Figure 8b. A/D Conversion with Dither
Figure 9 Impulse Response of Two-Point Finite Memory Filter
Figure 10. Impulse Response of Two-Point Infinite Memory Filter.

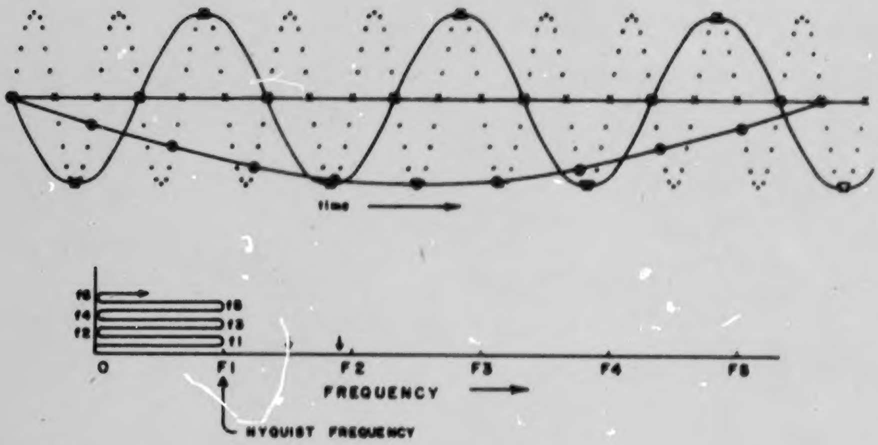

time
f6
f4
f2
f5
f3
f1
0
F1
F2
F3
F4
F5
FREQUENCY
NYQUIST FREQUENCY

FIG. I SAMPLED ALIASING

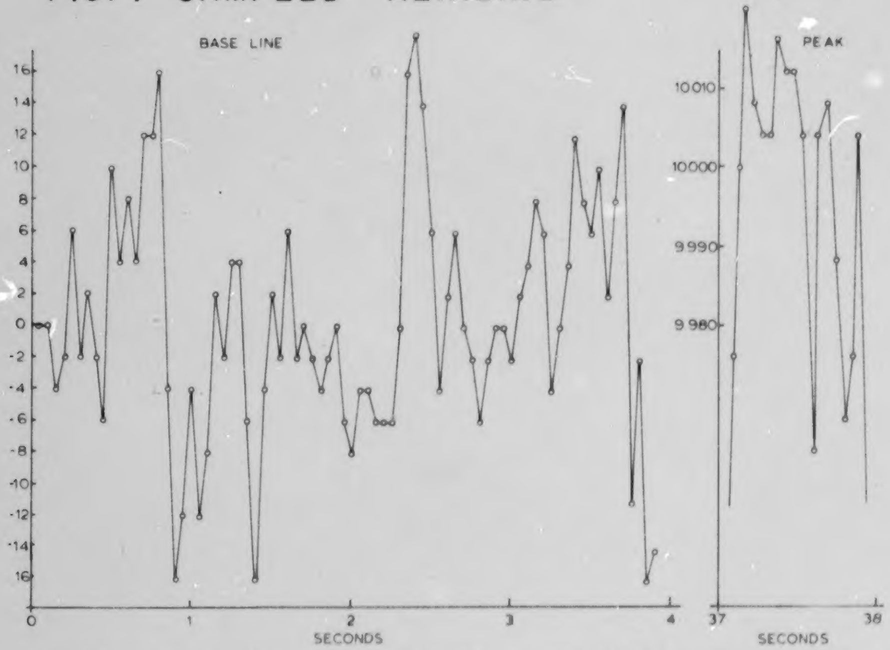


FIG. 2a 1KH$_z$ BANDWIDTH CHROMATOGRAM

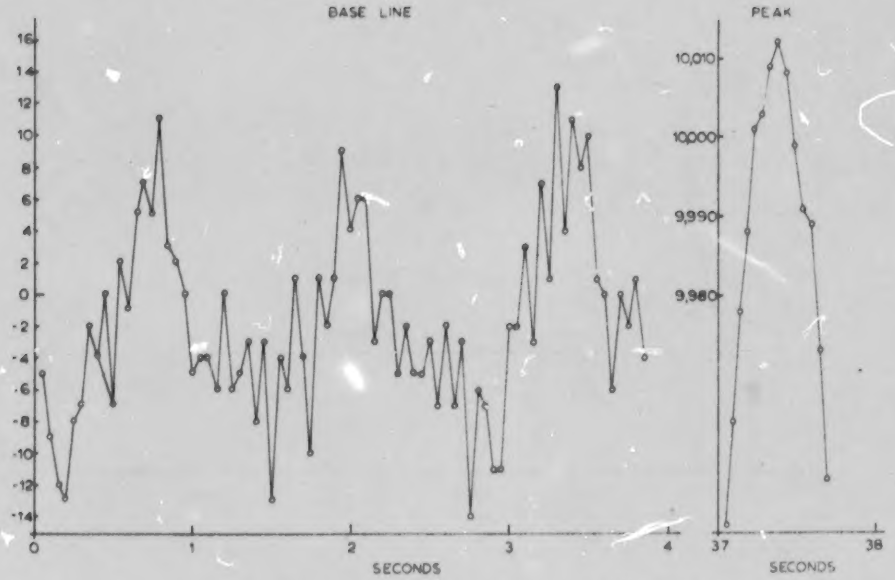


FIG. 2b 10 H_z BANDWIDTH CHROMATOGRAM

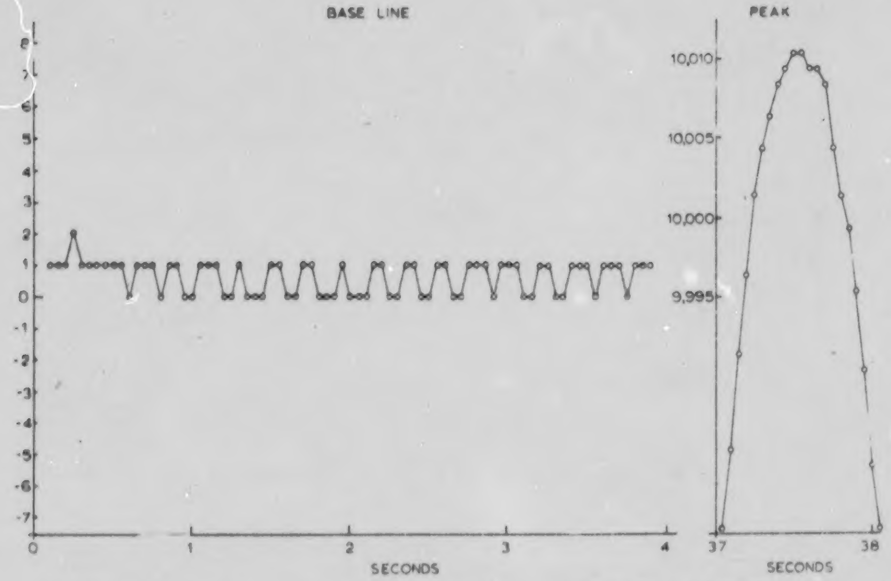


FIG. 2c 1 H_z BANDWIDTH CHROMATOGRAM

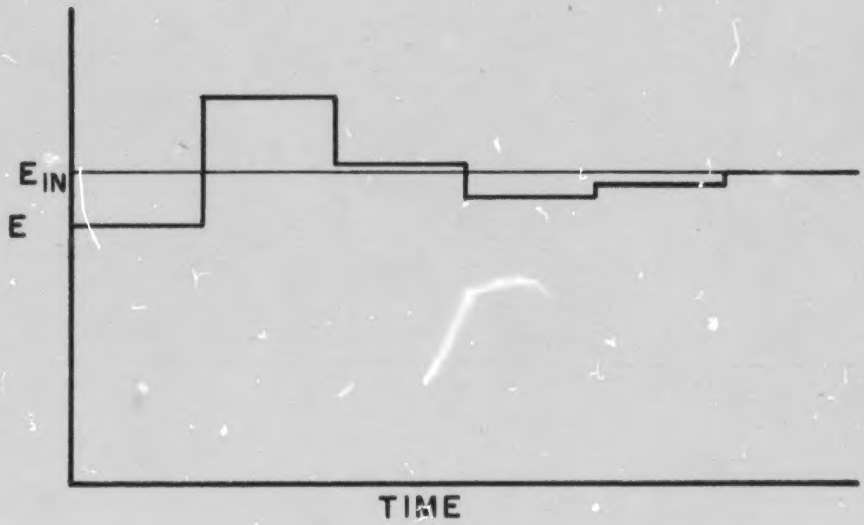


FIG. 3a SUCCESSIVE APPROXIMATION A/D CONVERTER

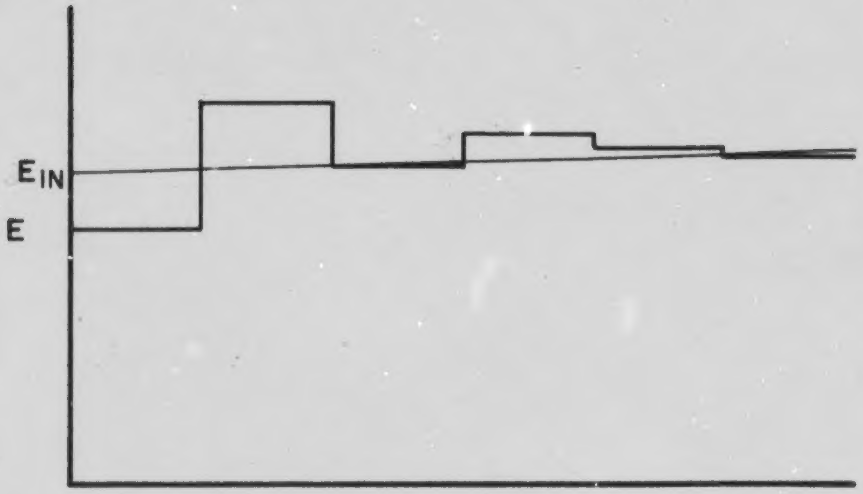


FIG. 3b SUCCESSIVE APPROXIMATION CONVERTER WITH DRIFT

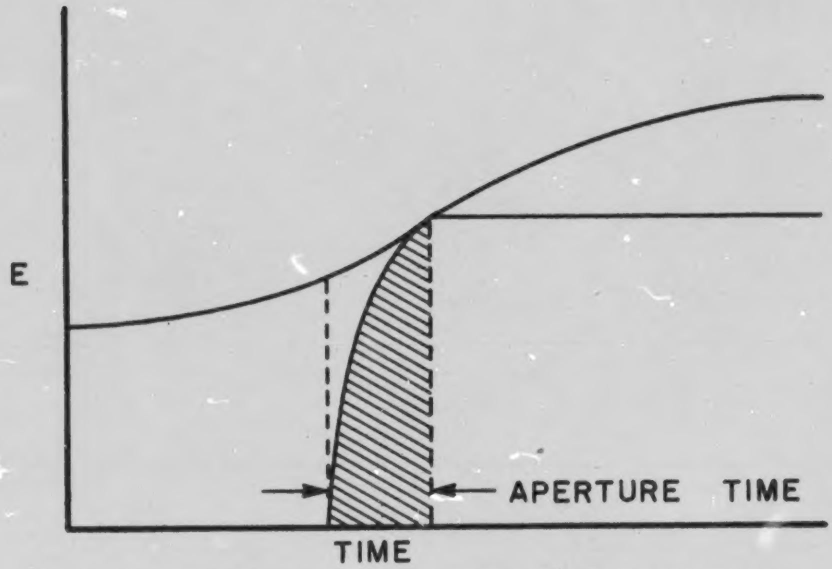

E
APERTURE TIME
TIME

FIG. 4 SAMPLE AND HOLD OPERATION.

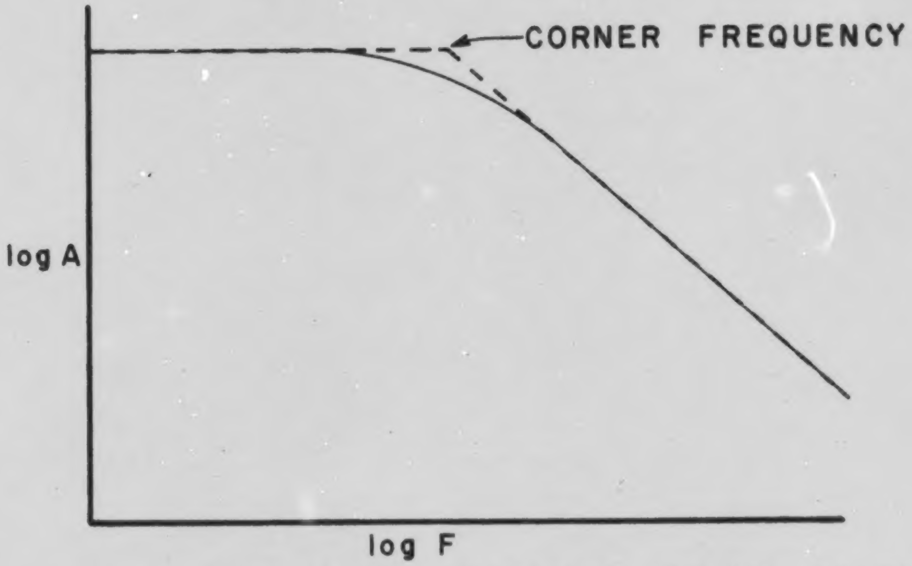


FIG. 5 AMPLITUDE RESPONSE OF AN RC FILTER.

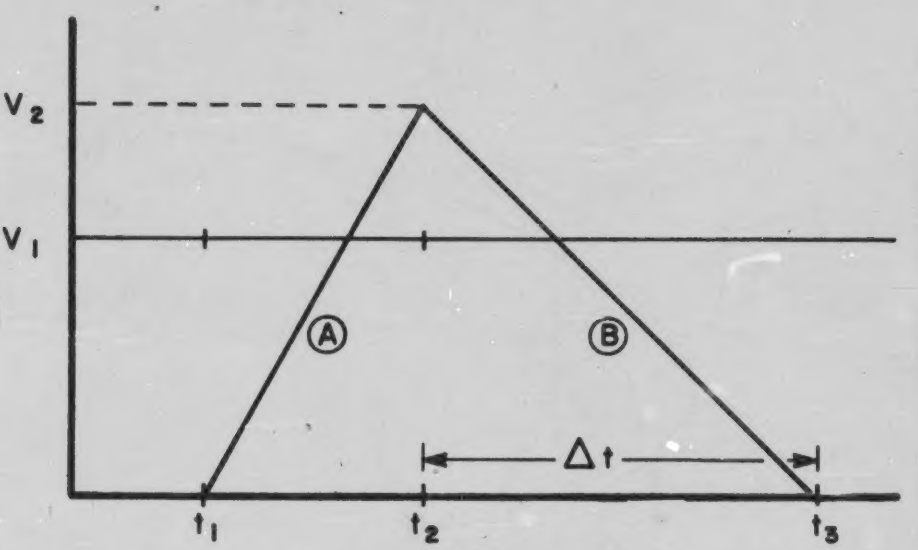

V₂
V₁
Ⓐ
Ⓑ
Δt
t₁
t₂
t₃

FIG. 6 – DUAL SLOPE INTEGRATING ANALOG-TO-DIGITAL CONVERTER

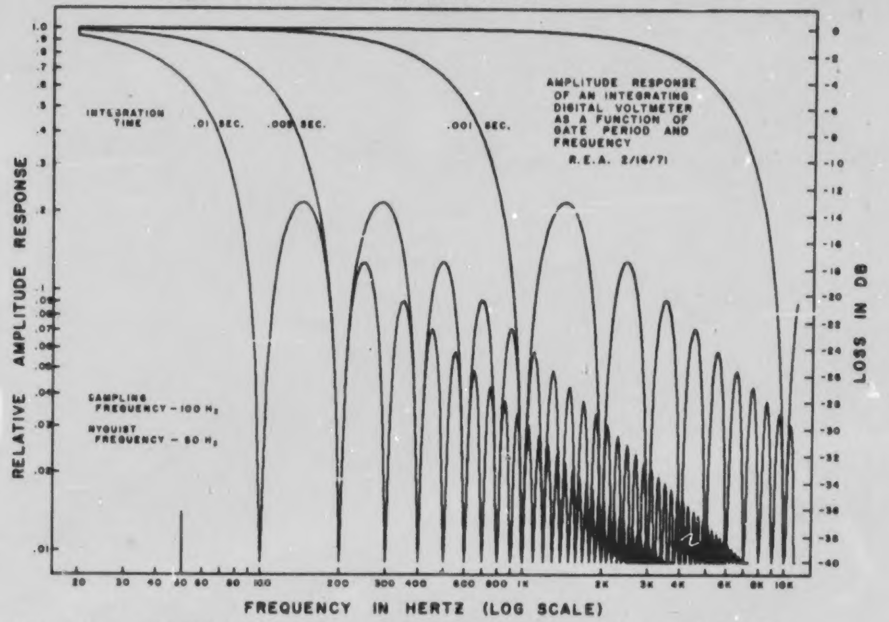


FIG. 7 AMPLITUDE RESPONSE OF GATED INTEGRATOR.

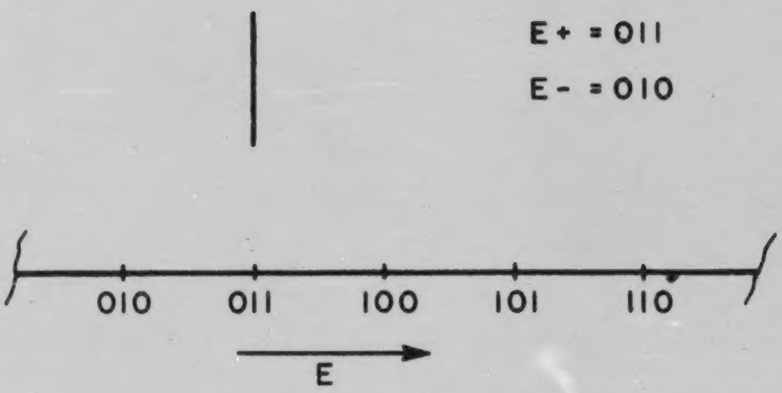

E+ = 011
E- = 010
010
011
100
101
110
E

FIG. 8a QUANTIZATION ERROR.

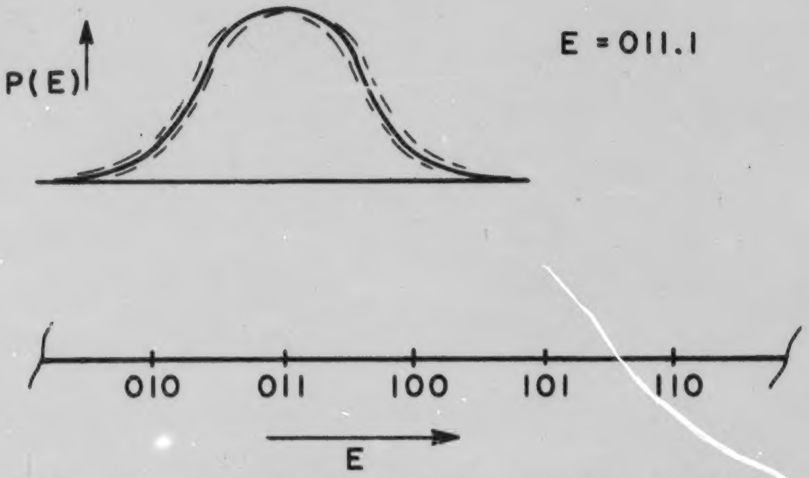

P(E)
E = 011.1
010
011
100
101
110
E

FIG. 8b A/D CONVERSION WITH DITHER.

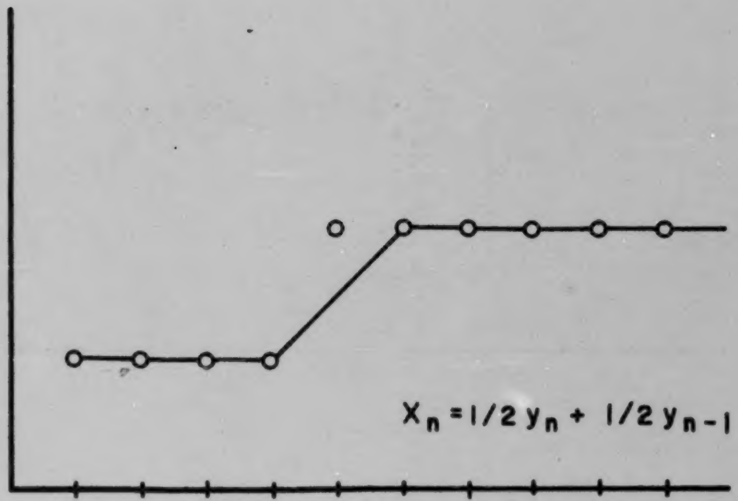


FIG. 9 IMPULSE RESPONSE OF TWO-POINT FINITE MEMORY FILTER.

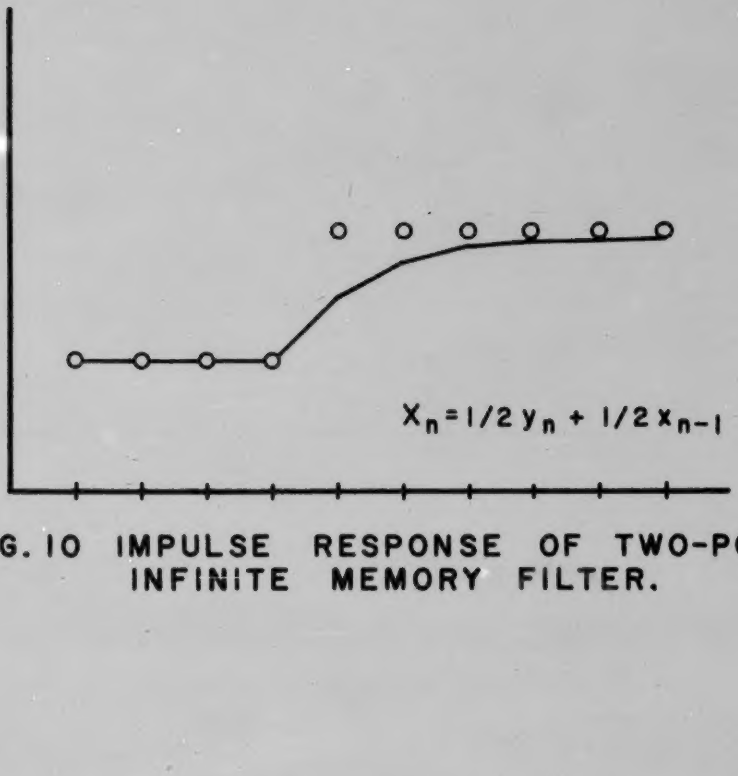


FIG. 10 IMPULSE RESPONSE OF TWO-POINT INFINITE MEMORY FILTER.

Computer Assisted Valuation of Error

by Ralph D. Nelson, Jr.
Chemistry Department
Middlebury College
Middlebury, VT 05753

F. Richard Ellenberger
Value Planning
General Electric Corp.
Schenectady, N.Y. 12301

Mark R. Ellenberger
Chemistry Department
Middlebury College
Middlebury, VT 05753

Abstract: A brief treatment of the statistical moments of data, estimation of imprecision, and propagation of error using numerical analysis is given. A method for taking input data with their estimated imprecisions and propagating the imprecision through a computer program to arrive at a final result and its estimated imprecision is developed. The theory is developed for cases where the imprecisions in the data are uncorrelated or independent of one another. The method allows for distributions of error which are nonGaussian, but is illustrated with a normal case. Current programs can easily be modified to incorporate the error analysis subroutine, with some nuisance in notation and input-output format. Computation time will increase by a factor of ten for error propagation computations.

Introduction: The computer systems now available often free us from major concerns of minimizing computation time. Attention can now be turned to using the computer for a more complete use of the data, namely, for the determination of the uncertainty in the final result. The six or eight figure numbers produced by the computer printout are completely unrelated to the number of digits to which the number should be "rounded off", and a casual analysis of error can be quite misleading. The error propagation method described below can give the researcher a well founded estimate of the precision of his results. At the same time it provides a tool for determining which factors in the data are leading to the most serious imprecisions.

Instruction in the use of computers to reduce data taken in undergraduate laboratories is incomplete unless error propagation is included. Students can best appreciate the problem of "weak links" in the data chain and the danger of overconfidence in computed results if the estimate of imprecision is carried out along with the main computation. The method below was designed with the beginner in mind. While it can be expanded to serve more sophisticated needs, it can be applied almost as an added packaged subroutine to most simple programs used in the physical chemistry laboratory.

Approaches: Various techniques have been developed to cope with the problem of error estimates. (Hahn and Shapiro) First: the Monte Carlo technique requires that a number of program runs be made, each with a different set of input data chosen from a statistically representative set of data values chosen about the mean values of the input parameters. The distribution of results can then be analyzed to determine the imprecision of the mean value. The method is rather lengthy and not easy to insert into current programs.

Second: A step by step algebraic determination of the propagated error may be used. (Strobel) Some skill at taking derivatives is required, and the computer program may need considerable expansion to evaluate the resulting equations. Third: A numerical differentiation about the mean value of a computation can be used in a Taylor series expansion about that point to get statistical moments about the mean value. This combines the best of the two methods above and results in a more convenient modification to current programs.

Theory of Moments: The statistical moments of a set of N values, x_i, about zero are defined (Kendall and Stuart) as

$$\langle 1 \rangle \qquad M_j = \frac{1}{N} \sum_{i=1}^{N} x_i^j$$

where j is an integer larger than zero. The first of these is called the mean, $\bar{x} = M_1$. The moments of the set of data (in this case we mean repeated measurements of the same experiment, differing in value only because of experimental imprecision) about the mean are particularly useful for unimodal distributions which have a monotonic decrease of probability of observation as the possible value gets farther away from the value most likely to be observed, the mode. In such cases the shape of the distribution can often be defined to satisfaction in terms of the first few moments about the mean. These are defined as

$$\langle 2 \rangle \qquad L_j = \frac{1}{N} \sum_i (x_i - \bar{x})^j$$

Here L_1 is zero by the definition of $\bar{x}$. The variance, L_2, is the square of the standard deviation, σ. The lack of symmetry about the mean is measured by L_3. A normalized quantity called the skewness is defined as

$\beta = L_3/L_2^{3/2}$, while kurtosis is defined as $K = L_4/L_2^2$.

The latter is a normalized measure of peakedness, L_4, that is, how sharply the distribution peaks near the mean relative to the height in the "wings" far from the mean.

The M values are somewhat easier to compute, but the L values are simply derived from them.

$$\langle 3 \rangle \qquad L_2 = M_2 - M_1^2$$

$$\langle 4 \rangle \qquad L_3 = M_3 - 3M_2M_1 + 2M_1^3$$

$$\langle 5 \rangle \qquad L_4 = M_4 - 4M_3M_1 + 6M_2M_1^2 - 3M_1^4$$

Estimation of Moments for Data: Rather than determining the statistical moments for each parameter (at each value in its range) experimentally, one can make some reasonable estimates about the limits of expected error for each parameter. For each parameter, use the mean value $\bar{x}$ (the value observed), the lowest expected value A (smaller than 99.9% of the values expected to be observed if replicate observations were made), and the highest expected value B (larger than 99.9% of the anticipated replicates). These can be used with a simple and general distribution form which is readily related to the statistical moments. (Ellenberger)

This polynomial distribution h[x] is related to the normal distribution of the standardized normal variate z, which has a distribution g[z], by the relation

$x = az^3 + bz^2 + cz + d$, and note that $g[z] = \frac{e^{-z^2/2}}{2\pi}$.

Specification of the coefficients and relation to the L and M values is most easily accomplished through some more relations, derived from A, B, and $\bar{x}$ above. The distribution span S=B-A is related to the moments by

$$\langle 6 \rangle \qquad S = 6\sqrt{15L_2/(15-2\beta)}$$

which reduces to 6σ for symmetric distributions. The skew ratio $R=(\bar{x}-A)/S$ is defined in a complex manner, but for $0.3 < R < 0.7$ it can be approximated by the form

⟨7⟩ $R = \frac{1}{2} - \frac{2}{9}B$ From these we can get

⟨8⟩ $L_2 = S^2\left[\frac{1}{36} - \frac{3}{40}\left(\frac{1}{2} - R\right)^2\right]$

⟨9⟩ $L_3 = \frac{9}{2}\left(\frac{1}{2} - R\right)L_2^{3/2}$ for $0.3 < R < 0.7$, and then get

⟨10⟩ $M_2 = L_2 + M_1^2$ ⟨10⟩ $M_3 = L_3 + M_1(3M_2 - 2M_1^2)$.

The coefficients relating x to z are defined as

⟨11⟩ $a = \frac{5}{108}\left[\sqrt{4(1-2R)^2+1} - 1\right]$

⟨12⟩ $b = \frac{S}{18}(1-2R)$

⟨13⟩ $c = \frac{S}{12}\left[3 - \sqrt{4(1-2R)^2+1}\right]$

⟨14⟩ $d = A + RS$.

This procedure may be skipped, of course, if a symmetric distribution ($L_3=0$ or $R=0.5$) is assumed, in which case the standard deviation may be estimated directly.

Theory of Propagated Error: If a computed result y depends on several input variables ($x_1, x_2, ..x_m, ..$) whose errors are uncorrelated, then the statistical moments of y can be calculated using the functional dependence y=f[x's] of the result on the input parameters and the known or estimated statistical moments of the x's. This is accomplished (Scarborough) by expanding the function in a multivariate Taylor series. Using the notation $\bar{x}_m$ for the mean of the mth parameter, and expanding to second order only, we get

⟨15⟩
$$y = f[x's] = f[\bar{x}_m's] + \sum_m \left(\frac{\partial y}{\partial x_m}\right)_{\substack{x_k = \bar{x}_k \\ \text{for } k \neq m}} (x_m - \bar{x}_m)$$
$$+ \frac{1}{2}\sum_m \left(\frac{\partial^2 y}{\partial x_m}\right)_{\bar{x}_k} (x_m - \bar{x}_m)^2 + 2\sum_m \sum_{n>m} \left(\frac{\partial^2 y}{\partial x_m \partial x_n}\right)_{\bar{x}_k} (x_m - \bar{x}_m)(x_n - \bar{x}_n).$$

Computer Assisted Valuation of Error

The first statistical moment about zero is then

⟨16⟩ $$M_{1y} = \bar{y} = f[\bar{x}_m's] + \frac{1}{2}\sum_m \left(\frac{\partial^2 y}{\partial x_m^2}\right)_{\bar{x}_k} L_{2m} .$$

The second moment about the mean of y is

⟨17⟩ $$L_{2y} = \sum_m \left(\frac{\partial y}{\partial x_m}\right)^2_{\bar{x}_k} L_{2m} + \sum_m \left(\frac{\partial y}{\partial x_m}\right)_{\bar{x}_k} \left(\frac{\partial^2 y}{\partial x_m^2}\right)_{\bar{x}_k} L_{3m} .$$

and the third and fourth moments about the mean are

⟨18⟩ $$L_{3y} = \sum_m \left(\frac{\partial y}{\partial x_m}\right)^3_{\bar{x}_k} L_{3m}$$ and

⟨19⟩ $$L_{4y} = \sum_m \left(\frac{\partial y}{\partial x_m}\right)^4_{\bar{x}_k} L_{4m} + 6 \sum_m \sum_{n>m} \left(\frac{\partial y}{\partial x_m}\right)^2_{\bar{x}_k} \left(\frac{\partial y}{\partial x_n}\right)_{\bar{x}_k} L_{2m} L_{2n} .$$

Thus, given a known function and the statistical moments for the input parameters, the appropriate partial derivatives may be obtained analytically and used in equation form in the computer program to get the moments of the computed result. An alternative route makes use of a subroutine to determine the derivatives numerically by varying the input parameter values about their mean values. The latter technique is chosen for further development here, since it requires no analytic differentiation with the consequent added programming. It can be used with most current programs if minor alterations in input and output format are made and a simple and general subroutine is added.

Numerical Differentiation: Evaluate $y=f[x_m's]$ at several values of one of the variables x_m, these values spaced equally by h_m about the mean $\bar{x}_m$, keeping the other variables at their mean values $\bar{x}_k$. Differences between the y values thus obtained are labelled

⟨20⟩ $$\Delta y_i = y[x_k, \bar{x}_m - (i+1)h_m] - y[\bar{x}_k, \bar{x}_m - ih_m]$$ and

⟨21⟩ $$\Delta^2 y_i = \Delta y_{i+1} - \Delta y_i$$ or, in general,

⟨22⟩ $$\Delta^p y_i = \Delta^{p-1} y_{i+1} - \Delta^{p-1} y_i .$$ The partial derivatives are related to these differences by infinite series, but a truncation of these series to the first few terms gives satisfactory results. For the first few derivatives, the truncated series approximations are (Scarborough)

$$\langle 23\rangle \left(\frac{\partial y}{\partial x_m}\right)_{\bar{x}_k} = \frac{1}{h_m}\left\{\frac{\Delta y_{-1} + \Delta y_0}{2} - \frac{1}{6}\left(\frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2}\right)\right\}$$ and

$$\langle 24\rangle \left(\frac{\partial^2 y}{\partial x_m^2}\right)_{\bar{x}_k} = \frac{1}{h_m^2}\left(\Delta^2 y_{-1} - \frac{1}{12}\Delta^4 y_{-2}\right).$$ These are computed for each input parameter, using $h_m = \sqrt{L_{2m}/4}$ or some other convenient interval, and are combined as indicated by equations 16-19. If there are further computations where y is involved and error propagation is desired, these results can be incorporated in the next stage of computation.

Notes on Implementation: The program given in the Appendix calls a standard subroutine whenever there is an error propagation to be done in the main program. The calling statement replaces the line used for the equation computation in the old (unmodified, without the error propagation) program. The equation is now listed in a second subroutine which is used in the numerical differentiation. The equations thus listed are in a standard form, y(N3)=some function of x(I)'s. This simplifies the form of the standard subroutine, although it has the unfortunate effect of obscuring the meaning of the program to the casual reader. That is, variables which in the original program were easily distinguished because different letters were used to represent them will have to be changed to x(1),x(2),etc.

The program given here uses only the mean and the standard deviation of input parameters and computes only these two measures for the output results. Thus only equation 16 and the first term of 17 are used. The readin format of the old program and the scaling and printout of the old program must be changed to accomodate the new symbolism of x's and y's and to make room for the standard deviation next to the mean. A more complete moment analysis would require more space and computer time, but would be equally straightforward to setup.

Appendix: The following program is written in the BASIC language for the Dartmouth Time-sharing Network.

Computer Assisted Valuation of Error

```
100 REM THIS PROGRAM CONVERTS THE READINGS OF A N
102  'IMPEDANCE BRIDGE TO PARALLEL R AND C DATA.
104  'PUT DATA IN AT LINE 3000 FF: FREQ(KHZ), D READING,
106  'C READING.         ***PROGRAMMER: R.D.NELSON-05753
110 LET M=8000*ATN(1)    'A CONSTANT
200 DIM X(30),E(30),M(5,5)
205 DIM Y(10)
210 READ N1 '#INPUT PARAMETERS WITH ERROR
215 READ N4 'NUMBER OF DATA SETS TO RUN THROUGH
220 FOR I=1 TO N1
230  READ X(I),E(I) 'MEAN, STANDARD DEVIATION
240  NEXT I
245 REM FREQ,D,C ARE INPUTS
250 LET N3=1 'CALC H=F*D
255 GOSUB 4000 'TO AVOID ERROR CALC, USE GOSUB 5000
260 LET N3=2 'CALC CP=C/(1+H↑2)
265 GOSUB 4000
270 LET N3=3 'CALC RP=(10↑6)/(M*F*D*CP)
275 GOSUB 4000
280 IF ABS(X(5)-X(3))<0.1*X(3) THEN 400
290 PRINT "LARGE CORRECTION TO CAPACITANCE"
400 PRINT "FREQ(KHZ)=";
402 LET N2=1
404 GOSUB 6020
410 PRINT "D READING=";
412 LET N2=2
414 GOSUB 6020
420 PRINT "C READING=";
422 LET N2=3
424 GOSUB 6020
440 PRINT "PARALLEL CAPACITANCE(PF)=";
442 LET N3=2
444 GOSUB 6000
450 PRINT "PARALLEL RESISTANCE(MEGOHMS)=";
452 LET N3=3
454 GOSUB 6000
456 PRINT
460 LET N4=N4-1
470 IF N4>0 THEN 220
500 STOP
3000 DATA 3,2
3010 DATA 1,0.001, 0.00015,0.00001, 210.567,0.002
3020 DATA 10,0.01, 0.13,0.04, 210.567,0.002
4000 LET Y=Y1=0 'SUBROUTINE FOR MEAN AND STD DEV OF CALCULATION
4010 FOR I=1 TO N1+N3-1  'EACH POSSIBLE PARAMETER
4020  LET E=E(I)/2  'SET INCREMENT SIZE
4030  LET X=X(I)-3*E
4040  FOR K=1 TO 5 'THE SET OF Y'S FOR ONE PARAMETER VARIATION
4050   LET X(I)=X+K*E
4060   GOSUB 5000
4070   LET M(K,1)=Y(N3)
4080   NEXT K
```

```
4090  FOR L=2 TO 5 'DIFFERENCES IN THE Y SET
4100   FOR K=1 TO 6-L
4110    LET M(K,L)=M(K+1,L-1)-M(K,L-1) 'SEE EQ 22.
4120    NEXT K
4130   NEXT L
4140  LET X(I)=X+3*E  'RESET THE X
4150  LET Y=Y+M(2,3)-M(1,5)/12 'SEE EQ 24.
4160  LET Y1=Y1+(M(2,2)+M(3,2)-(M(1,4)+M(2,4))/6)↑2 'SEE EQS 17,23
4170  NEXT I
4180 LET X(N1+N3)=2*Y+M(3,1)  'THE MEAN OF Y, SEE EQ 16
4190 LET E(N1+N3)=SQR(Y1) 'THE STD DEV OF Y
4200 RETURN
5000 ON N3 GO TO 5010,5020,5030
5010 LET Y(1)=X(2)*X(1)
5015 RETURN
5020 LET Y(2)=X(3)/(1+X(4)↑2)
5025 RETURN
5030 LET Y(3)=1000000/(X(5)*X(2)*X(1)*M)
5035 RETURN
6000 PRINT X(N1+N3);"+-";E(N1+N3)
6010 RETURN
6020 PRINT X(N2);"+-";E(N2)
6030 RETURN
9999 END
```

```
RUN      (on the Dartmouth Time-sharing Network)

CAVE       12 APR 71   14:57

FREQ(KHZ)= 1. +- 0.001
D READING= 0.00015 +- 0.00001
C READING= 210.567 +- 0.002
PARALLEL CAPACITANCE(PF)= 210.567 +- 2.00113 E-3
PARALLEL RESISTANCE(MEGOHMS)= 5061.33 +- 335.965

LARGE CORRECTION TO CAPACITANCE
FREQ(KHZ)= 10. +- 0.01
D READING= 0.13 +- 0.04
C READING= 210.567 +- 0.002
PARALLEL CAPACITANCE(PF)= 85.3291 +- 30.2851
PARALLEL RESISTANCE(MEGOHMS)= 1.75014 +- 0.671412

TIME: 0.458 SEC.  (includes compilation time)
READY
```

References:

F.R.Ellenberger, General Electric Information Document Number 69MSV100 (1969).

G.J.Hahn and S.S.Shapiro, Statistical Models in Engineering (Wylie, 1967).

M.G.Kendall and A.Stuart, The Advanced Theory of Statistics, Vol. 1 (Hafner Publishing Co.,1958).

J.B.Scarborough, Numerical Mathematical Analysis, (Johns Hopkins Press)

H.A.Strobel, Chemical Instrumentation, Chap. 2, (Addison Wesley, 1960).

Acknowledgement: Some of the time involved in preparing this work was provided by a National Science Foundation grant, GP-22829, to RDN.

1-30

CORRECTED FLUORESCENCE SPECTRA BY COMPUTER POSTPROCESSING*

V. Alan Mode, David H. Sisson, Carolyn A. Bailey

Lawrence Radiation Laboratory, University of California
Livermore, California 94550

INTRODUCTION

The availability of relatively low-cost, high-quality fluorescence instrumentation has enabled the routine use of fluorescence analysis by many laboratories. In quantitative analytical work, one need not be concerned with the spectral response functions of the instrument because analysis is performed at one wavelength and with an empirically determined calibration curve. However, if the instrument is used for qualitative analysis or for quantum efficiency determinations, a corrected spectrum is required. (A corrected spectrum is defined as one for which the observed intensity data have been corrected to reflect the spectral response of the exciting source, the optical system, and the photomultiplier-detection system.)

Every instrument, even from the same manufacturer, has a unique total spectral response. Therefore, the comparison of spectra between different instruments or with published data can be misleading or impossible. It is unfortunate that much of spectral data in the literature have not been corrected. Frequent reasons for this omission have been the very high cost of instrumentation for automatically producing corrected data, and the laborious effort involved in manually correcting data from more conventional instruments.

Several articles have been published on the correction of fluorescence spectra and the techniques for determining the calibration factors.[1-10] The grating efficiencies of the excitation and emission monochromators are essentially constant over their lifetimes. Once determined, the appropriate grating correction factors can be applied by means of electrical or mechanical circuits or by postprocessing of the data on a computer. The response function of the photomultiplier is also relatively constant and can be checked periodically with a standard lamp of known output. Like the grating corrections, the photomultiplier response corrections can be made during the postprocessing of the data.

Although systems can be designed that will make corrections during sampling, the electrical and/or mechanical systems required are complex, expensive, and require continual maintenance. If it is necessary to replace a photomultiplier tube, it may be necessary to recalibrate the entire correction system; a time-consuming operation. However, these response functions are easily handled by a computer and can be quickly changed to reflect changes in instrumentation.

To obtain a true corrected spectrum, it is also necessary to correct for the spectral distribution of the source. For post-processing of the data, a simple technique is required for determining the spectral distribution of the excitation source; this distribution must remain stable for several hours, and abnormalities (e.g., flicker or arc wander) must be quickly detectable. We have developed techniques to fulfill these requirements for postprocessing spectral data.

Our calibration and monitoring techniques combined with rapid postprocessing of the instrumental data has enabled us to produce high-quality, corrected spectral data with conventional instruments for a minimum capital expenditure.

INSTRUMENTATION

An Aminco-Bowman spectrophotofluorometer, Model SPF 4-8203, was modified to provide digital output for wavelength and intensity.[11] With the output from the wavelength digital voltmeter as a trigger, a logic circuit was designed to record the fluorescence intensity data on punched paper tape. A block diagram of the equipment is shown in Fig. 1 and a logic flow chart in Fig. 2.

Although the photomultiplier-microphotometer (Aminco-Bowman Model 10-26) has rated output of 1 V at 100% meter deflection, the output was found to be essentially linear up to 1.8 V (an 80% overrun of the meter indication). Thus the instrument ranges are effectively expanded, and data are provided which should not be lost. With a simple fifth level ASCII coding, it was possible to obtain four significant figures of data while recording only the three least significant figures. Therefore, intensity data from 0.000 to 1.999 were available with a minimum of punched output.

To produce the highest density of data per character punched, several arbitrary limitations were imposed on the system. A spectrum would always start at 200 nm and end at 799 nm (600 data points or 40 Teletype lines of 15 words each), and data would be recorded at a fixed increment of 1 nm. Under these conditions, it was necessary to generate wavelength-intensity pairs; intensity data alone were sufficient. It was necessary to ensure that the system would record only one intensity datum at each wavelength and would terminate recording if a datum was missed. In addition, the manual termination of a spectrum prior to the normal 799 nm had to be accomplished in such a way as to retain tape format. A 15-word TTY line had to be completed and the associated control characters punched to ensure compatibility with the computer hardware and software.

Our logic circuit (Fig. 2) provides all the necessary features to ensure proper collection of the data at high speed and at a high tape density (71.4 data characters/100 characters punched). Detailed wiring diagrams of the logic and punch circuits are available from the authors.

The required instrumental correction factors are either measured directly (e.g. the excitation correction) or are independently determined and then manually processed to give a proper input file (e.g., the photomultiplier response). Once determined, the correction data became a part of the computer library until replaced or updated to reflect instrumental changes.

A full description of the calibration of the system and the determination of the correction factor library has been published.[13]

DISCUSSION

The LRL Computation Center is composed of a number of digital computers, including three CDC 6600's and two CDC 7600's. The machines are operated in a modified "time-share" environment[12] with the user having access to any of the five machines from his local TTY via a network system involving PDP-6 and -8 computers (Fig. 3).

The existence of this large digital computation facility, and its relatively inexpensive local access, eliminated consideration of a small dedicated computer for our application. Three data transmission systems were considered: punched cards, magnetic tape, and paper tape. The equipment and interface costs plus the necessary manual transportation to the Computation Center (∿ 0.25 miles) eliminated punched cards and magnetic tape as the primary mode of data collection. (It should be noted that both cards and tape are used for long-term storage.)

In addition to the low cost for equipment and the interface, punched paper tape held the great advantage in a time-share environment of direct transmission to the central processor via an ASR-33 Teletype. It was relatively easy to design the paper tape interface so that the data collection rate was limited by instrumental parameters rather than by the punch. Because our data has essentially infinite life once recorded (as opposed to data used for instrument control, for example), the slow data transmission rate of the Teletype is not a factor. The code was written so that once a tape was started, the central processing computer provided all necessary Teletype start-stop instructions.

Our coding was done on the CDC 6600 using FORTRAN IV language. The logic flow chart for the processing program is shown in Fig. 4. The retention of the raw, uncorrected data file (File A in the figure) allows immediate recovery if the user selects an inappropriate option or wishes to serially output data only partially corrected by one of the library functions. Because the uncorrected data are still the primary results of the experiment, they are the only data retained long-term on cards or magnetic tape.

There are several important features of postprocessing of fluorescence data. One is the ability to see the effects of each correction function on the data. This can be particularly important when working near the maximum limits of the instrument where one correction factor may greatly distort the data. For educational purposes, the individual contribution of each function can be displayed and the overall importance of the correction technique demonstrated.

If an output format is fixed, one will surely find occasions when a different format is the only one that will properly display a result. Using the postprocessing technique, we have essentially complete freedom in displaying the data, wavelength in nanometers or wavenumbers, output in quanta or intensity.

While the general application of the technique of postprocessing has been previously discussed by Drushel[14] and others, we believe our specific use of digital postprocessing of fluorescence data demonstrates the general usefulness of the technique to a large number of users in education, research, and industry.

---

* This work was performed under the auspices of the U.S. Atomic Energy Commission.

[1] W.H. Melhuish, *J. Phys. Chem.*, *64*, 762(1960).

[2] C.A. Parker and W.T. Rees, *Analyst*, *85*, 587(1960).

[3] C.E. White, M. Ho, and E.Q. Weimer, *Anal. Chem.*, *32*, 438(1960).

[4] W.H. Melhuish, *J. Opt. Soc. Amer.*, 52, 1956(1962)

[5] C.A. Parker, *Anal. Chem.*, *34*, 502(1962).

[6] R.J. Arqauer and C.E. White, *Anal. Chem.*, *36*, 368(1964).

[7] G.K. Turner, *Science*, *146*, 183(1964).

[8] R.F. Chen, *Anal. Biochem.*, *20*, 339(1967).

[9] P. Byrom and J. B. Hudson, *Talanta*, *15*, 714(1968).

[10] L. Costa, F. Grum, and D.J. Paine, *Appl. Opt.*, *8*, 1149(1969)

[11] V.A. Mode and R.A. Thomas, *Rev. Sci. Instrum.*, *40*, 1241(1969).

[12] University of California Lawrence Radiation Laboratory, Livermore, Livermore Time Sharing System Manual M-026. (Available from the A.E.C. Division of Technical Information Extension, Oak Ridge, Tenn.)

[13] V.A. Mode, R.A. Thomas, and D. H. Sisson, *Rev. Sci. Instrum.*, *41*, 1714(1970).

[14] H.V. Drushel, A.L. Sommers, and R.C. Cox, *Anal. Chem.*, *35*, 2166 (1963).

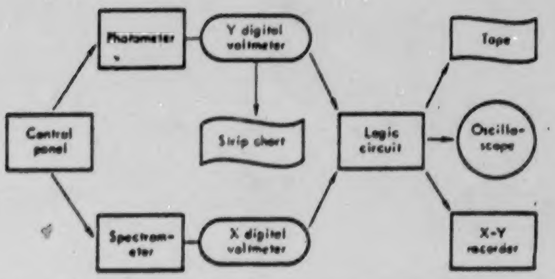

Photometer
Y digital voltmeter
Tape
Control panel
Strip chart
Logic circuit
Oscilloscope
Spectrometer
X digital voltmeter
X-Y recorder

FIG. 1. Block diagram of the modified spectrophotofluorometer.

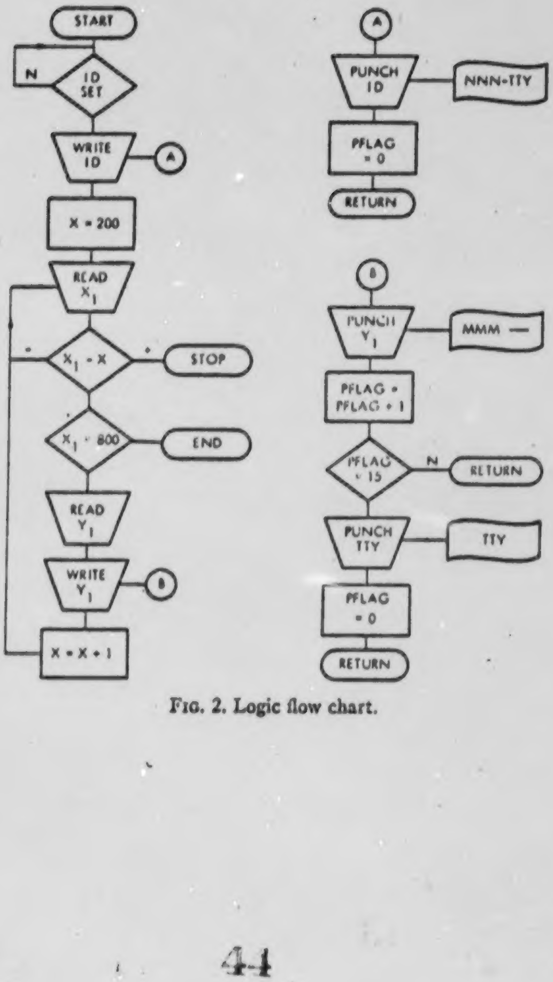

START
N
ID SET
WRITE ID
A
X = 200
READ X_1
$X_1 - X$
STOP
$X_1 = 800$
END
READ Y_1
WRITE Y_1
B
X = X + 1
A
PUNCH ID
NNN-TTY
PFLAG = 0
RETURN
B
PUNCH Y_1
MMM —
PFLAG = PFLAG + 1
PFLAG = 15
N
RETURN
PUNCH TTY
TTY
PFLAG = 0
RETURN

FIG. 2. Logic flow chart.

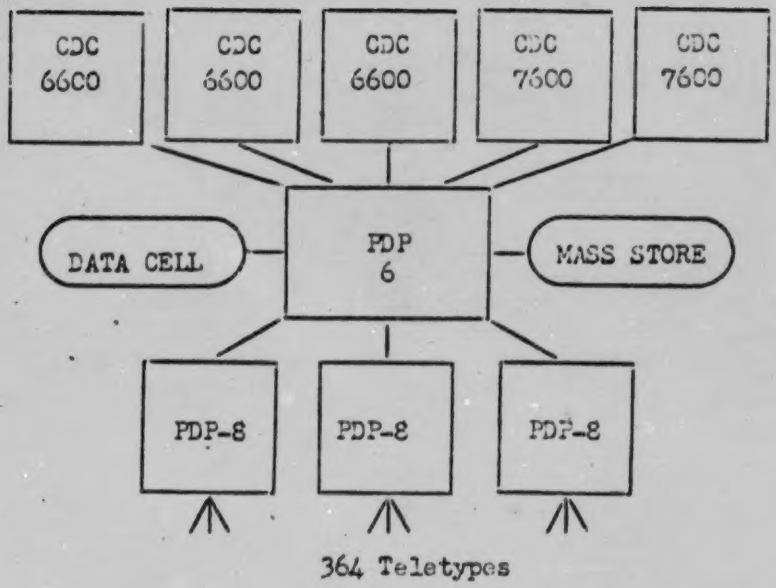

CDC 6600
CDC 6600
CDC 6600
CDC 7600
CDC 7600
DATA CELL
PDP 6
MASS STORE
PDP-8
PDP-8
PDP-8
364 Teletypes

Fig. 3 LRL - Livermore "Octopus" system

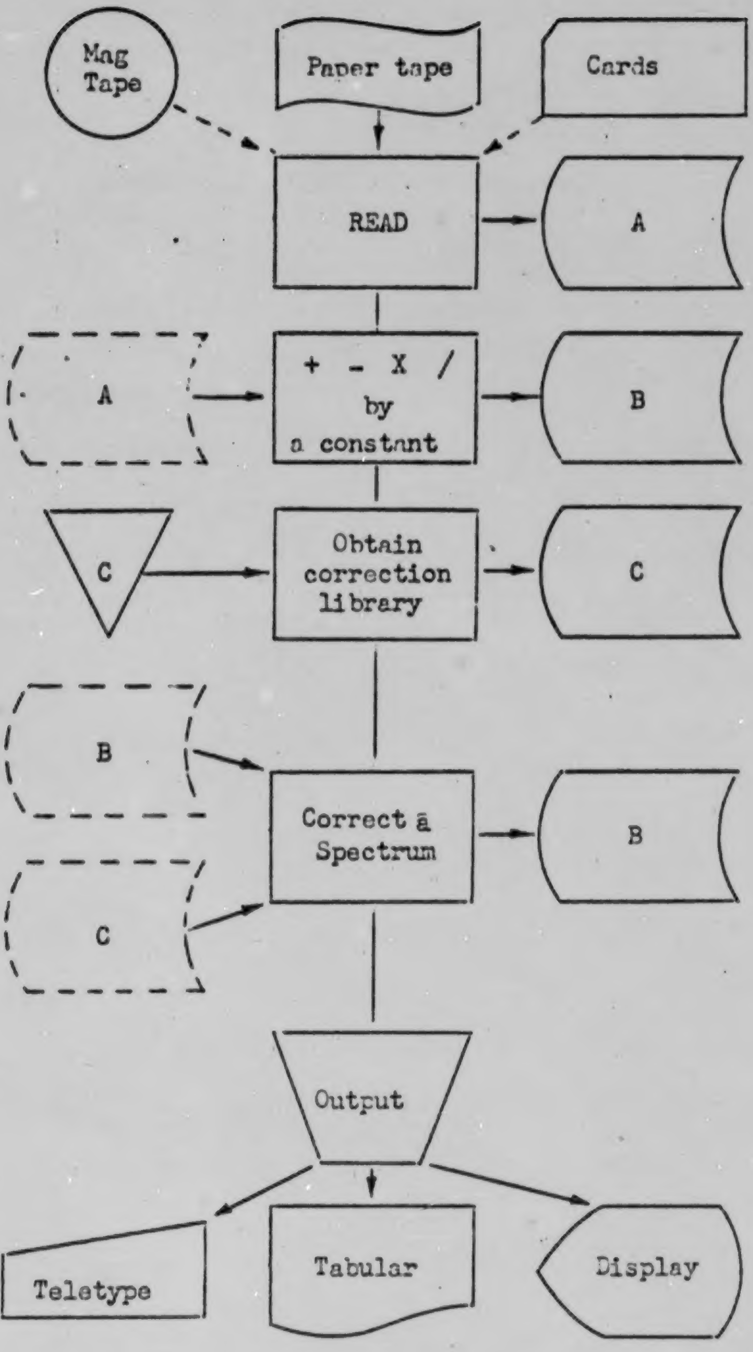

Mag Tape
Paper tape
Cards
READ
A
A
+ - X /
by
a constant
B
C
Obtain correction library
C
B
Correct a Spectrum
B
C
Output
Teletype
Tabular
Display

Fig. 4 Postprocessing flow chart.

LEAST SQUARES AND STATISTICS IN CHEMISTRY

John Overend
Molecular Spectroscopy Laboratory
University of Minnesota, Minneapolis, Minn.

In all branches of chemistry we frequently encounter the problem of determining empirical values for the parameters in the mathematical model of a physico-chemical phenomenon. Usually we have more experimental data than there are parameters to be determined and we resort to statistical analysis based on the principle of least squares. The basic idea is quite simple; suppose we have a linear relationship between the explaining parameters, x_j, and the observables, y_i, i.e.,

$$y_i = \sum_j J_i^j x_j \qquad (1)$$

where the coefficients, J_i^j, are determined by the model and are known in advance. The problem is to determine the best estimates of the explaining parameters such that the calculated values of the observables, $\bar{y}_i$, match closely the experimental values, y_i. The criterion applied is that the sum of the squares of the errors,

$$S = \sum_i \varepsilon_i^2 = \sum_i (y_i - \sum_j J_i^j x_j)^2 \qquad (2)$$

be a minimum. Differentiating eq (2) partially with respect to each x_j and setting the derivative equal to zero we obtain the set of normal equations

$$\sum_i J_i^j y_i = \sum_i \sum_k J_i^j J_i^k x_k \tag{3}$$

and the solution of these gives the least squares estimates of the explaining parameters, $\bar{x}_j$. The normal equations of eq (3) are conveniently written in matrix notation as

$$\underset{\sim}{J}'\underset{\sim}{y} = \underset{\sim}{J}' \underset{\sim}{J} \underset{\sim}{x}. \tag{4}$$

Here $\underset{\sim}{J}$ is a matrix of the coefficients J_i^j (cf. eq (1)), $\underset{\sim}{y}$ is a vector of the observables and $\underset{\sim}{x}$ a vector of the explaining parameters. In matrix form the least squares problem is nicely set up to be programmed for a computer.

If we have relatively more confidence in some of the experimental data, these can be given greater weight in the least squares analysis by requiring that the "weighted sum of errors squared", i.e., $\sum_i p_i \varepsilon_i^2$, be a minimum. In matrix notation this gives a set of normal equations

$$\underset{\sim}{J}' \underset{\sim}{P} \underset{\sim}{y} = \underset{\sim}{J}' \underset{\sim}{P} \underset{\sim}{J} \underset{\sim}{x} \tag{5}$$

where $\underset{\sim}{P}$ is a diagonal matrix, the elements of which are the weights associated with the corresponding observables. The least squares estimates of the explaining parameters are given by

$$\underset{\sim}{x} = (\underset{\sim}{J}' \underset{\sim}{P} \underset{\sim}{J})^{-1} \underset{\sim}{J}' \underset{\sim}{P} \underset{\sim}{y}. \tag{6}$$

If the relationship between the observables and the explaining parameters is nonlinear, $y_i = f_i(x_j)$, it is linearized by expansion in a Taylor series and the J_i^j coefficients of eq (1) are then the partial derivatives $\partial y_i/\partial x_j$. In that case the least squares is reduced to the linear problem except that, rather than determine the explaining parameters directly, the least squares procedure is used to determine the corrections to be applied to an initial (assumed) set of explaining parameters, x_j^0. We define Δy_i as

$$\Delta y_i = y_i - f_i(x_j^0) \tag{7}$$

and determine the corrections, x_j (to be added to x_j^0) from the equation

$$\Delta \underset{\sim}{x} = (\underset{\sim}{J}' \ \underset{\sim}{P} \ \underset{\sim}{J})^{-1}\underset{\sim}{J}' \ \underset{\sim}{P} \ \Delta \underset{\sim}{y} \tag{8}$$

which is analogous to eq (6). Since the problem is nonlinear it is necessary to repeat the least squares calculation iteratively until the calculated corrections Δx_j become arbitrarily small.

The least squares determination of the parameters in a problem has an additional dividend; we obtain also estimates of the statistical dispersions (the uncertainties) and the correlation coefficients of the parameters. We state without proof that the square matrix $(\underset{\sim}{J}' \ \underset{\sim}{P} \ \underset{\sim}{J})^{-1}$, when multiplied by an appropriate coefficient c, gives an estimate of the variance-covariance matrix, Σ. The diagonal elements of Σ are the squares of the statistical dispersions of the corresponding explaining parameters,

$$\Sigma_{jj} = \sigma^2\{x_j\}$$

and the offdiagonal elements of Σ give the correlation coefficients,

ρ_{jk},

$$\Sigma_{jk} = \sigma\{x_j\}\sigma\{x_k\}\ \rho_{jk}. \qquad (10)$$

The estimation of the coefficient c, through which we obtain the variance-covariance matrix, is a recurrent problem. It is useful to consider two cases; the first is that in which the model of the chemical problem implied by eq (1) is relatively good and the statistical errors originate in the uncertainty in the physical measurement of the observables, y_i. If the random error in each measurement, $\sigma\{y_i\}$, can be estimated from the analysis of the experimental conditions, c is determined by

$$c = (1/f) \sum_i p_i \sigma^2\{y_i\} \qquad (11)$$

where f is the number of statistical degrees of freedom, i.e., the number of data points less the number of explaining parameters. A familiar example of this class of situation is the analysis of a simple rotation-vibration band of a diatomic molecule. The wavenumbers of the individual rotational lines in the band are the observables and there is a small random error associated with the measurement of each due to instrumental noise and the difficulty of picking out the center of the line. The observed wavenumbers of the lines are then fitted by least squares to an equation of the form

$$\nu_m = \nu_{vib} + (B' + B'')m + (B' - B'')m^2 + \ldots \qquad (12)$$

and the explaining parameters ν_{vib}, $(B' + B'')$ and $(B' - B'')$ are determined. Eq (12) is an excellent model with sound quantum mechanical foundations and the only source of random error in the derived parameters is that arising in the experimental determination of the wavenumbers of the lines in the spectrum.

The second case we consider is the estimation of the coefficient c if the observables are determined with relatively high precision but the model is marginally adequate to describe the physical phenomenon, usually through limitations in the theory. As an example of this situation consider the empirical relationship of the bond length, r_e, in a diatomic molecule to the stretching force constant, k_e, through Badger's rule[1] which may be stated as

$$r_e = (C_{ij}/K_e)^{1/3} + d_{ij} \qquad (13)$$

where C_{ij} and d_{ij} are empirical parameters and are assumed to depend only on the row of the periodic table in which the atoms i and j lie. In general r_e and K_e may be determined with negligible statistical error and the origin of the uncertainty in the derived parameters C_{ij} and d_{ij} stems from the approximate nature of Badger's rule. If we take the experimental values of r_e and K_e for a number of molecules which are expected to have the same values of C_{ij} and d_{ij} we may treat the r_e as observables and use least squares to obtain the best estimates of the Badger's rule parameters. To determine the variance-covariance matrix for these it seems reasonable to use a value for the coefficient c which reflects the overall quality of the fit of eq (13) to the experimental data. It is usual, therefore, to estimate c by

$$c = S^2/f \qquad (14)$$

where f is the number of degrees of freedom and S^2 is the sum of the squares of the differences between the experimental r_e values and those calculated from the experimental values of K_e and the least squares estimates of C_{ij} and d_{ij}.

An important point to note in the least squares analysis of experimental data arises if there is some intermediate reduction of the data before the final determination of the explaining parameters. Suppose we have, as before, a set of primitive observables, $\underset{\sim}{y}$, and a final set of explaining parameters, $\underset{\sim}{x}$, but that, rather than fit the $\underset{\sim}{x}$ directly to the $\underset{\sim}{y}$, we first reduce these to an intermediate set of parameters, $\underset{\sim}{z}$, and from these determine the $\underset{\sim}{x}$. This particular problem cropped up in our own work[2] on anharmonic force constants from the spectroscopic energies of vibrational states which, in some cases we first reduced to the anharmonic frequencies, ω_s and the anharmonic coefficients, $x_{ss'}$. The data reduction scheme may be diagrammed as follows:

$$\begin{tikzcd}
\underset{\text{(vibrational energies)}}{\underset{\sim}{y}} \arrow[rr, "J_{yx}"] \arrow[dr, "J_{yz}"'] & & \underset{\text{(anharmonic force constants)}}{\underset{\sim}{x}} \\
& \underset{(\omega_s,\ x_{ss'})}{\underset{\sim}{z}} \arrow[ur, "J_{zx}"'] &
\end{tikzcd}$$

In the direct adjustment of the $\underset{\sim}{x}$ to the $\underset{\sim}{y}$, we have

$$\underset{\sim}{J}_{yx}'\ \underset{\sim}{P}_y\ \Delta\underset{\sim}{y} = \underset{\sim}{J}_{yx}'\ \underset{\sim}{P}_y\ \underset{\sim}{J}_{yx}\ \Delta\underset{\sim}{x} \tag{15}$$

where $\mathbf{P}_y$ is the diagonal weight matrix attached to the primitive data, $\mathbf{y}$. Now, since by definition

$$\Delta\mathbf{y} = \mathbf{J}_{yz}\,\Delta\mathbf{z} \tag{16}$$

and

$$\Delta\mathbf{z} = \mathbf{J}_{zx}\,\Delta\mathbf{x},$$

$$\Delta\mathbf{y} = \mathbf{J}_{yz}\mathbf{J}_{zx}\,\Delta\mathbf{x}$$

and, since also by definition

$$\Delta\mathbf{y} = \mathbf{J}_{yx}\,\Delta\mathbf{x},$$

$$\mathbf{J}_{yx} = \mathbf{J}_{yz}\mathbf{J}_{zx}. \tag{17}$$

Substituting eq (16) and (17) in eq (15),

$$\mathbf{J}_{zx}{}'\mathbf{J}_{yz}{}'\mathbf{P}_y\mathbf{J}_{yz}\,\Delta\mathbf{z} = \mathbf{J}_{zx}{}'\mathbf{J}_{yz}{}'\mathbf{P}_y\mathbf{J}_{yz}\mathbf{J}_{zx}\,\Delta\mathbf{x} \tag{18}$$

and, if we compare this with the form of the normal equations used when $\Delta\mathbf{x}$ is computed directly from $\Delta\mathbf{z}$, i.e.,

$$\mathbf{J}_{zx}{}'\mathbf{P}_z\,\Delta\mathbf{z} = \mathbf{J}_{zx}{}'\mathbf{P}_z\mathbf{J}_{zx}\,\Delta\mathbf{x}, \tag{19}$$

we see that

$$\mathbf{P}_z = \mathbf{J}_{yz}{}'\mathbf{P}_y\mathbf{J}_{yz}. \tag{20}$$

And so we conclude that, for consistency in the statistical treatment of data, whenever we pass through an intermediate stage, we must carry through the proper weight matrix. It is interesting to note, that, apart from a trivial scale factor, the weight matrix $\mathbf{P}_z$ is the inverse of the variance-covariance matrix for the z parameters.

In practical least squares calculations in which one attempts to adjust a relatively large number of parameters to a few experimental data, even though there may be more data than parameters, this does not guarantee that all the parameters are determinable. Such difficulties become manifest when we attempt to invert the matrix J'PJ; if there are fewer data than parameters this matrix is singular and the inverse does not exist. Even if there are more data than parameters, the matrix may be extremely ill-conditioned and the results of inversion unreasonable - especially in the nonlinear least squares problem. Clearly there is then need to supply some constraint to stabilize the problem and the question is how to accomplish this in the most effective way. An idea with which we have experimented is the following; if the matrix J'PJ is singular its rank may be determined conveniently by diagonalizing and finding the number of nonzero eigenvalues. If it is nonsingular but ill conditioned it will not have zero eigenvalues but some will be very small. We may regard the diagonalization of the normal-equation matrix as the transformation to a new set of uncorrelated parameters which are linear combinations of the original parameters of the problem, the coefficients of the linear combination being given by the elements of the corresponding eigenvector. The diagonalized matrix J'PJ is easily inverted by simply taking the inverse of each element and gives the variance-covariance matrix of the new uncorrelated parameters. The small eigenvalues correspond to large dispersions and indicate which linear combination of the original explaining parameters is ill-determined. The useful constraints to apply to the problem are then identified as ones which have a large projection on the eigenvector of the ill-determined uncorrelated parameter.

The use of statistical techniques in the analysis of the data obtained in chemical experiments has grown explosively since computers became readily available and it might at this time be appropriate to remember that these techniques deal only with random errors. They are inappropriate for handling systematic errors, mistakes and deficiencies in the model used to interpret the results and it is particularly important to guard against these possibilities - it is so easy to bury them in the computer analysis of the data. It is important, therefore, to examine the final fit of the model to the original data, preferably by plotting a graph or alternatively, by examining carefully the differences between the observed and calculated values of the observables and checking that these fall in a reasonably random pattern with no systematic trends. The computer has made data analysis systematic and simple but it is no substitute for a careful examination of the physics and chemistry of the problem.

1. R. M. Badger. J. Chem. Phys. 3, 710 (1935).

2. I, Suzuki, M. A. Pariseau and J. Overend, J. Chem. Phys. 44, 3561 (1966).

OFF-LINE AUTOMATION SYSTEM FOR A RAMAN SPECTROMETER

Bernard J. Bulkin and Arthur Noguerola

Hunter College of the City University of New York

695 Park Avenue, New York, N.Y. 10021

The Raman effect is characterized by poor signal to noise ratios. In addition, spectra often appear as weak signals on high backgrounds. Because of these problems, some time ago we began to consider automation of our Raman instrument for digital data acquisition, signal averaging, and computer assisted data reduction. This program has now made considerable progress.

Most instrument automation systems feature on-line data acquisition using a variety of computers. Scherer (1) has constructed such an automation system for a Raman instrument using an IBM 1800 computer, and Bernstein et al.(2) have used a PDP-8 computer in an on-line system. For several reasons, we have chosen to use off-line automation of our Raman spectrometer, in which a hard-wired system controls the spectrometer, executing commands read sequentially from paper tape, with the data being punched out by teletype on to paper tape. Data reduction is done subsequently using a Nova computer in the laboratory.

The choice of an off-line system was made based on the following considerations: 1) The Raman spectrometer is a very stable system. Our laser power fluctuates very little even over the course of a 16 hour run, and there is generally no system deterioration over long time periods unless it involves sample degradation. 2) There is little decision making to be done during the course of a Raman spectrum. The operator can set all instrument parameters and these need not be changed over long time periods. Thus there seemed to be little need to provide the decision-making power of a computer. 3) For those spectra where digital data acquisition and signal averaging could provide definite improvements over analog methods, rather long times are required. These range from ca. 4 to 15 hours. Thus the time needed to print out the tape, and the time needed to load the output tape into a computer after the run, is very small compared to the time needed to acquire data. 4) It was felt desirable to carry out as much of the data reduction as possible within our own laboratory, rather than relying on outside computer facilities. With the off-line system and Nova computer, almost no danger exists of losing the data due to computer failure. Further, the system, once written, can run unmodified if this is desired, for several years. Our experience with computer centers has been that large periods of time are spent in minor revision of programs due to major system changes. 5) Since the computer is not needed during the experiment, it can do other things. In our case, the computer is used for control and data acquisition in a Fourier transform infrared spectrometer. This arrangement has the added advantage that Raman and infrared spectra are plotted using the same computer and plotter.

The discussion of the Raman automation system is divided into three parts: Control of the spectrometer, nature of the digital output, and data reduction.

Instrument Control: Figure 1 shows the functions of the controller built for us by the Digital Automation Co. of Trenton, New Jersey. Signals go from the control box to the

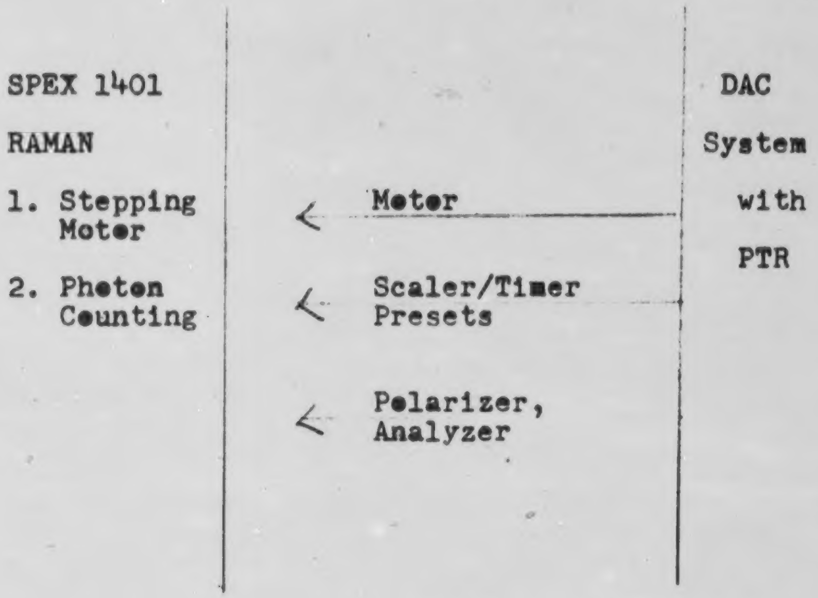

SPEX 1401
RAMAN
1. Stepping Motor
2. Photon Counting
Motor
Scaler/Timer Presets
Polarizer, Analyzer
DAC
System
with
PTR

Fig. 1. Schematic control of spectrometer

Spex Ind. spectrometer. The essential features with which the spectrometer must be equipped include a stepping motor and photon counting amplification. The motor makes a large number of steps per wavenumber increment, usually 72 steps/wavenumber, though by use of a gear box this can be increased to 72 steps/.001 wavenumber if this is desired. In addition to controlling the position of the spectrometer, presetting time and counts for a data collection cycle, and continuously counting and scanning, the controller will also rotate the polarization analyzer and half wave plate for automatic recording of depolarization ratios.

Commands are given to the spectrometer, through the controller, using a paper tape reader which is on the front of the control box. Any number of commands can be listed for sequential execution. The commands include: Position-set the spectrometer to a particular frequency: Increment- step through a spectrum, taking data for a preset length of time or to a preset count limit at stated intervals; Fixed- count for preset time or preset counts at a fixed frequency: Continuous- scan to a given terminal frequency, accumulating all counts in the region; Actuate- close any combination of three relays to rotate analyzer and/or half wave plate, and to turn off the laser at the end of the experiment; End- terminate data collection. The format of the commands is extremely simple. A typical command, for incrementally scanning from the current position, in the positive $\Delta\nu$ direction, to a terminal frequency of 200 cm^{-1}, in 1 cm^{-1} increments, with a preset count limit of 9 x 10^4 counts, is shown below.

I10200+00194.

For convenience, the exciting line is always set as 10000 on the wavenumber counter, thus avoiding the problem of negative values on the anti-Stokes side of the spectrum. The time is preset using a front panel switch.

Data Output: Figure 2 shows the functions of the controller in data output. The controller contains its own scaler and timer,

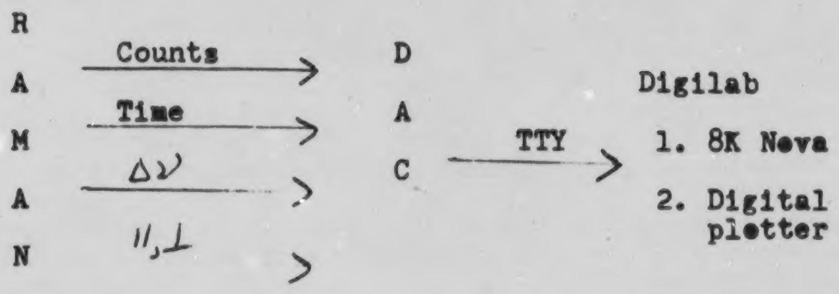


Fig. 2. Schematic output from spectrometer

which take data directly from the pulse height analyzer. There is not time constant circuitry, so raw data are obtained. An up-down counter counts the motor pulses for incrementing the wave-number counter. This has been found to be completely satisfactory. No shaft encoder has been used.

The format of the output tape is shown below. It contains the counts, time, and frequency shift, as well as relay closures, for each data point. Although the first three pieces of information are all punched as six digit numbers, this is really only necessary for the counts. The tape is formatted for easy manipulation by the computer. Two space characters are punched between each piece of information on a line, and carriage return/line feed characters separate lines. The tape is punched completely in ASCII code.

123456	123456	123456	1 CR,LF
counts	time	$\Delta\nu$	relays

Data Tape

Data Reduction: Once data acquisition is complete the tape is transferred to the Digilab FTS system with its Nova computer for data reduction. Although the ASCII coded tapes are quite long, they can be read into the computer in a short time using a high speed paper tape reader. Even if this is not available, only about twenty minutes is required via the teletype. This is not too severe a limitation, as the time needed to acquire the data is so long.

The computer as presently set up has 12K words of core memory, however we plan to convert to 8K of core memory and a 128K disc memory in the near future. This will greatly expand the possibilities for refinement of the data.

The computer immediately divides the counts by the time for each data point, converting to a counts/second format. Before plotting the data on the Houston Instruments DP-1 plotter, the computer enters a conversational mode. The number of data points, and their maximum and minimum intensity values, are printed on the teletype. The operator then is given the current values of the x and y scales for plotting, and can modify these via the teletype, based on the above information. Because spectra often have high backgrounds, the operator can ask to have the data offset to zero, and automatic scale expansion can be selected. The option of plotting points rather than a continuous line is also available. Finally, a binary tape of the data can be punched. This is much shorter than the original ASCII tape, and is generally the form in which spectra are stored.

After decisions as to plot format have been made, the spectrum is plotted. The data as plotted at this point are raw data, having no smoothing, either digital or analog. They often appear to be quite noisy for this reason.

Further data reduction options are also available. These include smoothing of data using either a moving average or parabola fit, band area determination for depolarization ratios, computation of the depolarization ratio and the uncertainty in that ratio using counting statistics and error theory, correction of intensities for radiometric and scattering phenomena, and so-called α, β plotting. This latter technique, developed by Scherer (3), is quite useful for separating totally symmetric modes from the spectrum if more than one symmetry species is Raman allowed. Frequency calibration is done using laser emission lines. If, as often occurs in Raman spectra, bands are present on a sloping background, the operator may have a particular region replotted with a sloping background subtraction to compensate for this. The scaling of the spectra is such that infrared spectra from the FTS system may be plotted directly on the Raman spectra.

Figure 3 shows the spectrum of CCl_4 in the 459 cm^{-1} region.

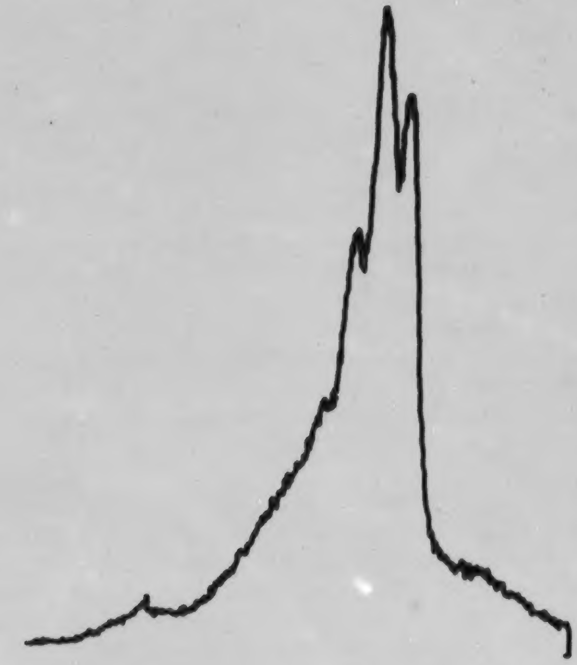


Fig. 3. ν_1 of CCl_4

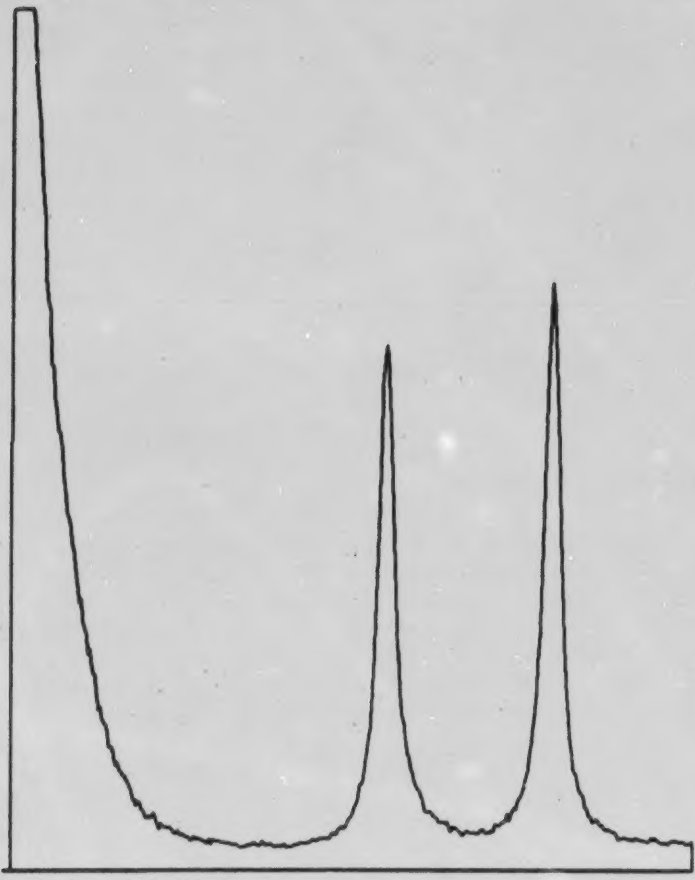


Fig. 4. Low resolution spectrum of CCl_4

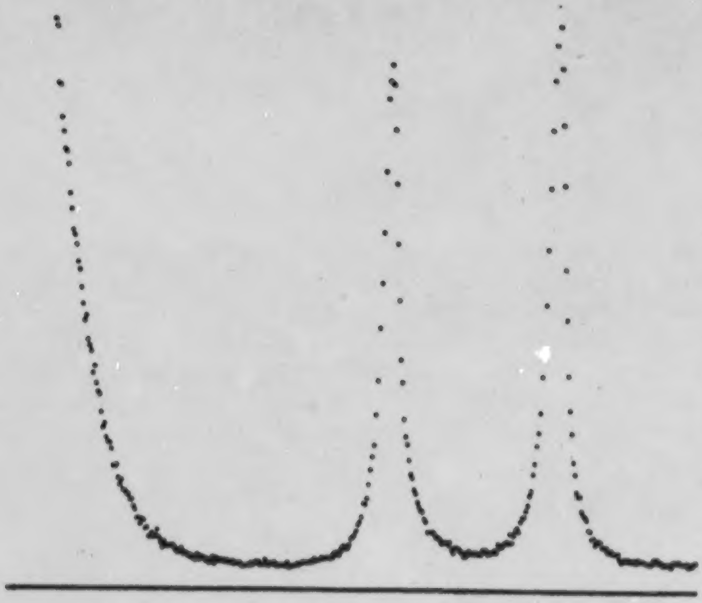


Fig. 5. Individual points of Fig. 4 plotted

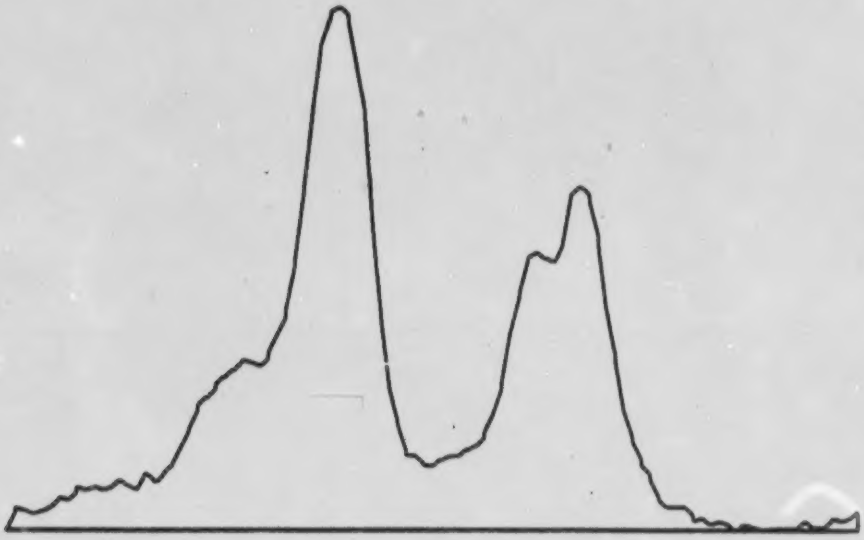


Fig. 6. Raman spectrum of p-azoxyanisole

This region was digitized at 0.1 cm^{-1} intervals for a time of 1 second at each point. A lower resolution spectrum of CCl_4 is shown in Figure 4, covering the region from 0-400 cm^{-1}. In Figure 5 this same spectrum is plotted using the point plotting mode. Figure 6 shows a region of the spectrum of p-azoxyanisole, from 1000-1100 cm^{-1} . In Figure 7 this same spectrum is replotted using background subtraction and automatic scale expansion.

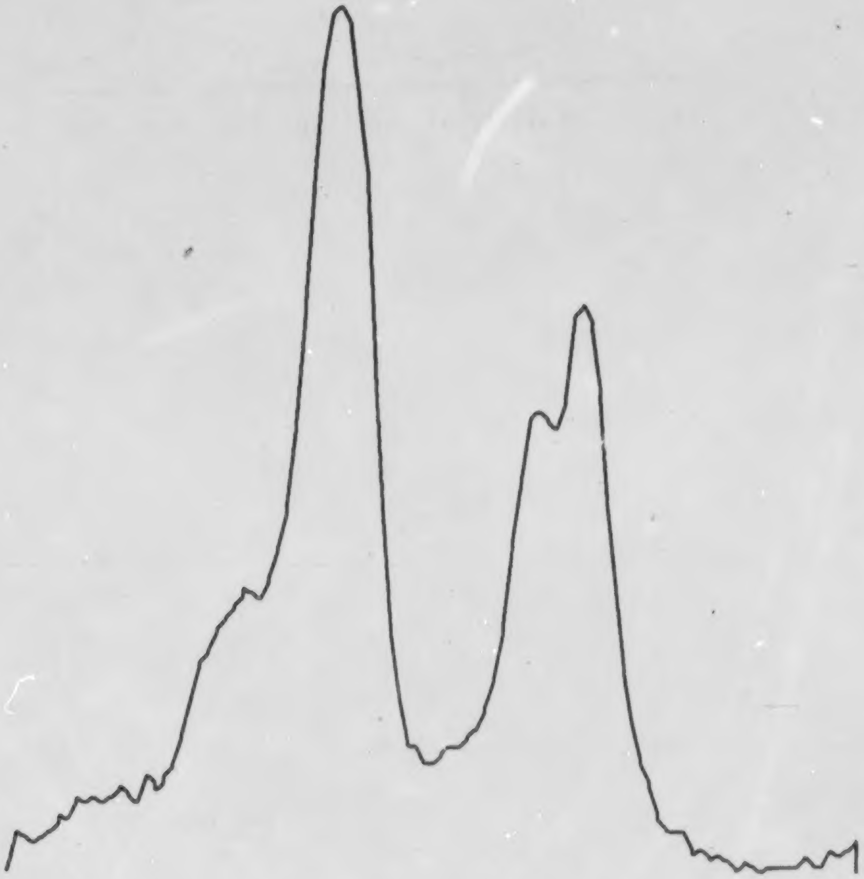


Fig. 7. Data from Fig. 6, replotted with background subtraction and scale expansion.

This system has been in operation for about 8 months, although certain data reduction features have been brought up more recently. Others are still in progress, such as slit function correction, line shape determinations, and least squares resolution of overlapping bands. During the period of operation, it has enabled us to acquire spectra which could not be obtained using analog techniques. These are cases in which extremely high backgrounds prevent data from being recorded in the analog system. Such instances are frequent in Raman spectra. The system has also opened certain research areas to us, particularly the study of Raman line shapes and errors in depolarization ratio measurements, which were not previously feasible using analog techniques. Finally, it has permitted to see our true experimental data for the first time, unmodified by analog ratemeter and recorder.

ACKNOWLEDGEMENT

This work was supported by a grant from the U.S. Army Research Office-Durham. We acknowledge several helpful conversations with C. Foskett and co-workers of Digilab, Inc., and the cooperation of Digital Automation Co. in construction of the system.

References

1. J. Scherer, Appl. Optics, 9, 1615 (1970).
2. E. Bernstein, W. Murphy, presented at the Symposium on Molecular Structure and Spectroscopy, Ohio State University, September 1969.
3. J. Scherer, private communication.

Computer Collection of Mass Spectrometric Ionization Efficiency Curves

Bruce R. Conard, Peter M. Fast, David L. Nordlund, and Paul W. Gilles
Department of Chemistry and Computation Center
University of Kansas, Lawrence, Kansas 66044

ABSTRACT

A data acquisition and equipment control system is being developed and used in the high temperature chemical laboratory at the University of Kansas. The central processing unit of the system is an 8K, 16-bit per word, general purpose, Hewlett-Packard 2116B digital computer.

The design specifications are given for the hardware of the system and for the software for one of the experiments, the collection of ionization efficiency curves with a Nuclide high temperature mass spectrometer. The hardware configuration adopted and the software developed for the data collection are described. Examples of data collected for H_2O^+ and $Cu_3Cl_3^+$ are shown. The advantages of automatic data acquisition over previous methods of manual collection are that (1) the data have improved precision and reliability, (2) the procedure may be exactly repeated, (3) the data collected are permanently stored in a very conveniently retrievable form, and (4) further data processing is readily initiated and accomplished in a reliable fashion.

INTRODUCTION

The high temperature chemical research program at the University of Kansas currently includes studying and characterizing vaporization reactions under equilibrium conditions, studying the kinetics of vaporization of structurally simple materials, and identifying solid phases by x-ray diffractometry. The principal apparatuses being used in the research are a Nuclide high temperature mass spectrometer, a quadrupole mass spectrometer manufactured by Electronics Associated, Inc., an Ainsworth semimicro recording balance-torsion effusion apparatus, and a Philips powder x-ray diffractometer.

In 1969, with a grant from the National Science Foundation, two identical Hewlett-Packard 2116B general purpose digital computers were purchased, one for the high temperature laboratory, the other for the microwave laboratory at the University of Kansas. The computers served as central processing units to data acquisition and equipment control systems currently in use.

The purpose of the work reported here was to interface the Nuclide mass spectrometer to the data acquisition and equipment control system for the purpose of collecting high temperature mass spectrometric ionization efficiency curves on vaporizing species. Presented elsewhere in this symposium are papers which report work performed on the computer control of the quadrupole mass spectrometer in the high temperature laboratory (1) and the development of the sister system in the microwave laboratory (2).

THE USE OF THE MASS SPECTROMETER

The use of the Nuclide high temperature mass spectrometer is grossly different from that ordinarily encountered in organic mass spectrometry because gaseous pressures are so low and because temperature dependences of ion intensities are often sought. The mass spectrometer is used to investigate chemical phenomena at high temperature. The rate of mass scan, for example, is usually on the order of one a.m.u. per 5 sec. at 100 a.m.u. The instrument is usually operated at a resolving power of 300-500. Many of the substances being investigated with the mass spectrometer have low volatility. For example, at 2000°K the vapor pressure of Ti_3O_5, in the Knudsen cell situated several inches from the ion source, is of the order of 10^{-7} atm.

The single-focussing mass spectrometer has a 60-degree sector, 12-inch radius magnet, an electron impact ion source, and both Faraday cup and electron multiplier detectors which are used in a mutually exclusive fashion.

A Knudsen cell containing a solid sample is introduced into a high vacuum system and is heated by radiation and electron bombardment. The temperature of the cell is measured by optical methods either with a manually operated Leeds & Northrup disappearing filament optical pyrometer or with a Leeds & Northrup No. 8642 automatic pyrometer. The gaseous species effuse out of the cell in a molecular beam which encounters a moveable 0.025-inch slit that when at a center position allows a portion of the beam to reach the ion source region. The neutral beam, after passing through various discriminating and focussing plates, is bathed with electrons having energies to within ±0.3 eV of a particular energy in the range 2-70 eV.

The ions formed by electron impact are accelerated through a potential of about 5000 V, are focussed through another series of plates, and then enter the magnetic field which allows a small ribbon of the ion beam to be focussed onto a detector. The current due to ions, termed the ion current, arriving at the electron multiplier detector, is multiplied through sixteen dynodes having a total gain of about 10^5 and measured by a Cary vibrating reed electrometer (VRE). The input to the electrometer is in the range of 0.001 to 30 volts, and the output is presented to a 10 mV strip chart recorder. Ion detection is also performed by amplifying the pulse from the last dynode of the multiplier and counting pulses with a Hewlett-Packard 5201L scaler.

The dependence of the ion current on ionizing electron energy constitutes ionization efficiency data, from which a chemist can obtain valuable information: 1) The appearance potential, that is, the minimum electron energy at which the ion is detected, can be obtained. 2) Identification of different ions having the same integral mass is aided by the value of the appearance potential. 3) Dissociation energies and bond energies for molecules can be obtained. 4) Fragmentation of a molecule caused by the impact of the ionizing electron with the molecule may in some cases be recognized by an analysis of the shape of the curve.

Some of this information can be obtained from ionization efficiency curves which have been collected under earlier manual operation, the procedure for which was as follows. The electron energy control potentiometer was set at 70 eV. A particular ionic species was focussed by manual variation of accelerating potential, magnetic field, and ion source focussing plates. For this measurement the potentials on the repeller plates were zero. Maximum ion current, as revealed by visual inspection of the VRE output on a meter or on the strip chart recorder, was taken to indicate that focussing had been achieved. The electron energy was then manually set at zero. By careful and very tedious manual operation the electron energy was varied approximately at 0.25 eV per second and the ion current traced on a strip chart recorder. Although this method was satisfactory for some purposes, it had limitations. Consider, for example, the possible problem of fragmentation in the Ti-O system.

Gilles and Hampson (3) established that the congruent vaporization of Ti_3O_5 occurs predominantly according to reaction (1),

$$Ti_3O_5\ (s) = TiO\ (g) + 2\ TiO_2\ (g). \qquad (1)$$

In characterizing this vaporization reaction and calculating thermodynamic quantities for the Ti-O system from mass spectrometric measurements it is imperative that the ion current observed for both TiO_2^+ and TiO^+ be proportional to their respective equilibrium partial pressures. If fragmentation occurs, however, according to reaction (2),

$$TiO_2 + e^- = TiO^+ + O + 2\ e^-, \qquad (2)$$

the TiO^+ ion current observed is not proportional to P_{TiO} in the Knudsen cell, but depends as well on P_{TiO_2}. Consequently, results derived from such measurements are not thermodynamically correct. In order to obtain reliable results, either the extent of fragmentation must be known for a particular ionizing electron energy, or the data must be obtained at electron energies below that energy required for fragmentation. Manually collected data have been too imprecise for the scientist to be able to detect the onset of fragmentation to produce TiO^+ from $TiO_2(g)$ in the presence of TiO^+ arising from TiO(g). Automatic data acquisition was undertaken to improve the precision of the measurements and thus to allow the experiment to be performed in such a way as either to avoid fragmentation or to correct for it.

SYSTEM DESIGN AND IMPLEMENTATION

Six general aspects were considered in designing the data acquisition and equipment control system (DAECS) to perform automated experiments, one of which was to be the collection of ionization efficiency data with the Nuclide mass spectrometer.

1. The Central Processing Unit. The choices for a central processing unit in a data acquisition system include a solid state sequencer which can provide control signals to various hardware devices; a large digital computer providing time-shared interaction through a remote terminal; and a small, dedicated, general purpose, digital computer providing real-time interaction with laboratory hardware.

For the automation of data acquisition in the high temperature chemical laboratory the central processing unit requirements are the following. The unit is to be dedicated because the rate of data collection is sometimes 100 per second, and prompt service is required. The acquisition of data and the control of hardware devices are to be flexible. Operator interaction with the central processing unit is to be easy and flexible. The acquisition and control functions performed during an experiment are to be performed simultaneously by the central processor. The servicing of hardware devices by the system is to be on a priority basis so that integrity of data is maintained.

2. Acquisition of Data. The data to be acquired from the mass spectrometer are the following: a 10-60 mV analog signal from the pyrometer, millivolt analog signals from voltages on various focussing plates in the ion source region, a millivolt analog signal characteristic of the ionizing electron energy, a 0-10 V analog signal from the gaussmeter, a 0-10 mV analog signal from the output of the vibrating reed electrometer, a digital signal from the output of the preamplifier-amplifier-scaler pulse counting chain, and a digital signal from a time-of-day clock.

3. Equipment Control. The control signals for the mass spectrometer equipment are of two types: 1) feedback control based on data acquired from the experiment, and 2) single-ended control for setting or changing an experimental variable. Other needed control signals are switch closures by relays and proportional voltages from a ±50 V digital-to-analog converter.

4. Operator Interaction. The possibility for intervention by the scientist during an experiment is desirable. The scientist is to be made aware of the status of the experiment by means of typed output, oscilloscope display, or X-Y or strip chart recorder displays. During the experiment the scientist should be able, for example, to revise initial parameters and direct experimental status changes.

5. Data Storage and Retrieval. The data storage and data retrieval are to be possible (a) by using the computer core memory, (b) by transmitting data to an on-line storage device, and/or (c) by transmitting the data to the sister system in the microwave laboratory or to the GE-Honeywell 635 computer for storage.

6. Data Processing. Data processing is to occur either simultaneously with data acquisition and equipment control or at a later time using data stored previously on a data storage device

such as magnetic tape or disc. Data processing of results by the on-line machine is to be kept to a minimum and is only to be performed if necessary to the five aspects considered above. Connections between the local computer and the 200 K, 36-bit-per-word, 1μs cycle time University computer are to be made to facilitate transmission of stored data for processing by the larger machine.

The specifications for the computer program to collect ionization efficiency curves under computer control are summarized as follows.

1. The ion intensity is to be obtained by pulse counting.
2. The electron energy is to be varied by means of a stepping motor.
3. The operator is manually to set the original electron energy and initialize several other parameters at execution time.
4. Multiple ionization efficiency collections (passes) are to be possible. A single collection is defined as a series of incrementing motor steps from the initial to the final electron energy setting.
5. The average of multiple collections is to be calculated immediately after the data from each pass are accumulated.
6. Rejection of a pass or portion of a pass by the operator is to be possible.
7. All data are to be stored on a mass-storage device.
8. The operator is to be able to interact with the data acquisition system to halt collection permanently or temporarily, to reject a pass, and to obtain a listing of the data as collected.
9. A real-time oscilloscope display of the signal average over all passes for a particular species is to be presented to the operator.

The hardware system satisfying the general criteria described above is shown in Figure 1 in which hardware devices are represented as rectangles. The central processing unit is the HP 2116B computer, a general purpose digital computer. It is a 16-bit-per-word computer with 8,192 words of core memory and a memory cycle time of 1.6 microseconds. The computer has a flexible input/output (I/O) system for connecting and servicing peripheral devices and allows various DAECS operations to be performed simultaneously. Thus the computer can overlap its acquisition, processing, storage, displaying, and controlling functions. The dedicated use and speed of the computer allow the development of new experimental techniques which require time measurements in microseconds.

In Figure 1 data acquisition devices are all represented to the left of the CPU HP 2116B computer. The arrow-headed lines connecting these devices to the computer show the direction of data flow. To the right side of the HP 2116B in Figure 1 are represented equipment control signals. Three control signals are also operator interaction devices, namely, the oscilloscope, the X-Y recorder and the strip chart recorder. The operator interaction devices are represented at the bottom of the diagram and the storage and retrieval devices at the top.

The ionization efficiency data are collected by the electron multiplier-preamplifier (ORTEC 109A)-amplifier (ORTEC 435)-scaler (HP 5201L) pulse counting chain. The computer controls the opening of the scaler gate.

Variation of the electron energy is achieved by a double-ended stepping motor on the electron energy control potentiometer shaft. The motor has 200 steps/revolution with a minimum of 4 msec delay between steps. The increment in each step is 0.05 eV. The computer controls the number and the rotational direction of the steps.

Initialization parameters are entered via the keyboard of a teletype. All parameters and data are stored on an HP 2870A disc unit having one million words of storage and average access time of 80 msec. The initialization parameters include (a) the name of the storage file on the disc, (b) the experiment title, (c) the bottom and top limits, in counts per second, of the scope display, (d) the total number of motor steps per pass, (e) the number of motor steps to be moved before initiating a scaler accumulation, and (f) the number of seconds the scaler gate is to remain open.

The oscilloscope display is obtained through a dual 8-bit 0-10V, D/A converter with a 25 msec refresh period and a Hewlett-Packard 140B oscilloscope. Operator interaction is possible by means of the teletype and by means of 16 toggle switches, which comprise a one word register located on the front panel of the HP 2116B. The options available via five of the sixteen toggle switches are: to suspend both data collection and motor control functions during a pass; to suspend functions at the end of a pass thus enabling the scientist manually to refocus the ion beam before the next pass is started; to list on the teletype the data collected for the current pass; and to terminate data acquisition and equipment control, returning control to the system executive program.

RESULTS

The data acquisition and equipment control system was used to collect the ionization efficiency curve for H_2O^+, mass 18, a background species with a partial pressure of about 10^{-9} atm. in the ion source region of the mass spectrometer. Figure 2 is a reproduction of the oscilloscope display after the collection of three passes had been completed. One division on the horizontal axis is equivalent to 10 eV; on the vertical axis, to 3000 counts per second. The total number of motor steps moved during the H_2O^+ data collection was 1206 at intervals of 6 steps about every one second. The initial electron voltage control potentiometer setting was 10 eV. The final setting was 70 eV. There were, therefore, 201 points displayed on the oscilloscope. Each point is the average of three passes. Each point of each pass was collected with the scaler gate time equal to one second.

The reproductions of the oscilloscope displays in Figure 3 (a), (b), and (c) show ionization efficiency curves for $Cu_3Cl_3^+(g)$. The collection was performed on $Cu_3Cl_3^+(g)$ vaporizing from a single crystal of CuCl(s) in an open graphite crucible which was heated to 215±10°C. At this temperature the partial pressure of Cu_3Cl_3 in the ion source region was about 10^{-10} atm. Figure 3(a) is the result after one pass, (b) is the average of the results of seven passes, and (c) is the average of the results of fifteen passes. For all passes the electron energy was stepped from 7 eV to 30 eV, in intervals of 0.2 eV about every 0.13 seconds. The gate on the scaler was open for 0.1 seconds for each accumulation. One division on the horizontal axis of the display is equivalent to 2.5 eV; on the vertical axis to 200 counts per second.

The scatter in the $Cu_3Cl_3^+$ collection after 15 passes, particularly at about 26 eV, was in part due to temperature fluctuations of the single crystal during the collection of several passes.

DISCUSSION

The automatic acquisition of ionization efficiency data resulted in the following advantages over the previous collection procedure.

1. The data have improved precision. The data collected for H_2O^+ at 50 eV and shown in Figure 2, for example, represent an accumulation of 31,500 total counts. The error associated with that number of counts is equal to 177 counts, or 0.6 percent of the observed quantity. This precision can be compared with a 4.3 percent error associated with the strip chart recorder measurement of the VRE analog signal at 50 eV for the same collection.

2. The collections of data are repeatable. Often, under previous manual operation, the electron energy control shaft was moved too much or too little and at a variable rate. For automatic data acquisition each digital ion intensity is associated with a discrete and definite electron energy.

3. The data are conveniently stored on a disc rather than on bulky and cumbersome strip chart paper.

4. The data are easily accessible in original form by the data acquisition system for further processing by the system, by the sister system in the microwave laboratory, or by the large University central computing facility. Because of this capability, the scientist may design the collection of data in new ways and may instruct the system to process the original data so that chemically meaningful results are available while the experiment is being performed. This kind of interaction was previously not possible because the manual data work-up delayed chemical results by days or weeks.

Future plans for the system include the collection of ionization efficiency data for TiO_2, TiO, and Ti from Ti_2O_3 and Ti_4O_7 samples at 1600° to 1750°C. as the samples vaporize to conguently vaporizing Ti_3O_5.

The data collection procedure will be modified to provide for computer-controlled refocussing between passes. Plans are also being developed for the determination of appearance potentials from ionization efficiency data by numerical methods.

Other data acquisition plans for the Nuclide mass spectrometer include mass scans, ion intensities as functions of temperature, shutter profiles, and pressure calibration measurements.

ACKNOWLEDGEMENTS

The authors are pleased to acknowledge the support of the National Science Foundation under its grant number GJ-332 which provided the computer, the associated data acquisition devices, and the personnel for hardware and software development; and of the U. S. Atomic Energy Commission under its contract AT(11-1)-1140 which provided the other equipment and personnel. They thank Mr. Dean E. Peterson for the cuprous chloride data.

REFERENCES

1. P. M. Fast, J. R. Castles, and P. W. Gilles. "Computer Control of a High Temperature Quadrupole Mass Spectrometer." Paper submitted to Conference on Computers in Chemical Education and Research, DeKalb, Illinois, (1971).

2. M. D. Harmony, D. K. Hendricksen, P. M. Fast, and D. L. Nordlund. "Data Acquisition and Equipment Control in Microwave Spectroscopy Using a Dedicated Digital Computer." Paper submitted to Conference on Computers in Chemical Education and Research, DeKalb, Illinois, (1971).

3. P. J. Hampson and P. W. Gilles. "High Temperature Vaporization and Thermodynamics of the Titanium Oxides. VII. Mass Spectrometry and Dissociation Energies of $TiO(g)$ and $TiO_2(g)$." Article submitted to the *Journal of Chemical Physics*, (1970).

DATA ACQUISITION AND EQUIPMENT CONTROL SYSTEM

HIGH TEMPERATURE CHEMISTRY LABORATORY

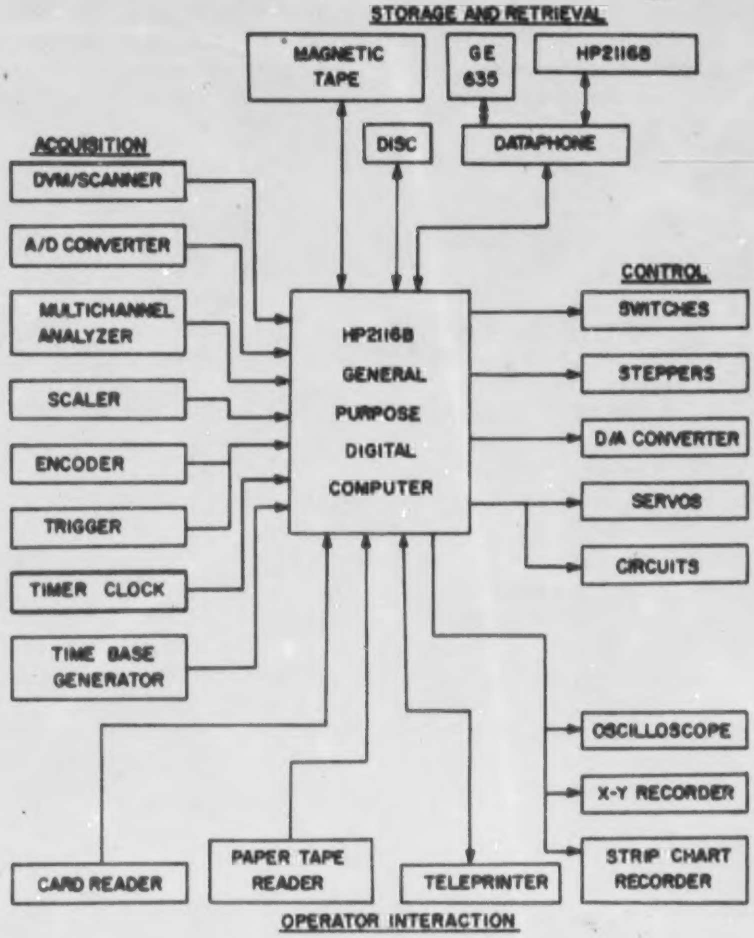

STORAGE AND RETRIEVAL
MAGNETIC TAPE
GE 635
HP2116B
DISC
DATAPHONE
ACQUISITION
DVM/SCANNER
A/D CONVERTER
MULTICHANNEL ANALYZER
SCALER
ENCODER
TRIGGER
TIMER CLOCK
TIME BASE GENERATOR
HP2116B GENERAL PURPOSE DIGITAL COMPUTER
CONTROL
SWITCHES
STEPPERS
D/A CONVERTER
SERVOS
CIRCUITS
OSCILLOSCOPE
X-Y RECORDER
STRIP CHART RECORDER
CARD READER
PAPER TAPE READER
TELEPRINTER
OPERATOR INTERACTION

Figure 1

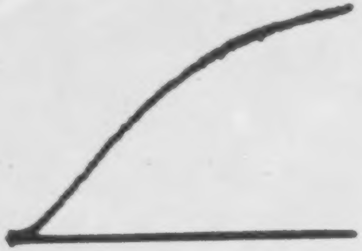


Figure 2. Ionization efficiency curve for H_2O^+ from 10 to 70 eV. Result of three passes.

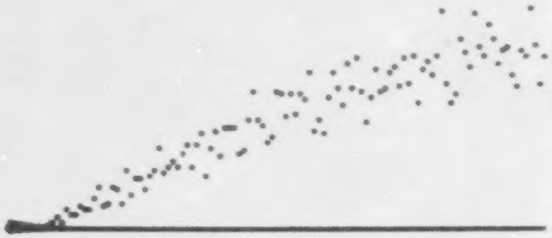


Figure 3(a). Ionization efficiency curve for $Cu_3Cl_3^+$ from 7 to 30 eV. Result of one pass.

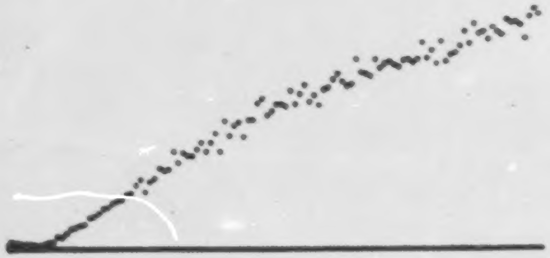


Figure 3(b). Ionization efficiency curve for $Cu_3Cl_3^+$. Result after 7 passes.

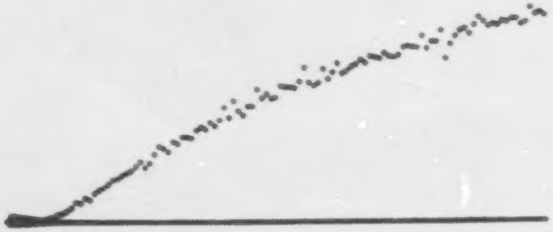


Figure 3(c). Ionization efficiency curve for $Cu_3Cl_3^+$. Result after 15 passes.

Overview - Session on Computer Assignment of Complicated Spectra.

Conference on Computers in
Chemical Education and Research
Northern Illinois University, July 19-23, 1971

W. T. Carnall
Chemistry Division
Argonne National Laboratory

The use of computers in simulating or analyzing complex spectra has been discussed in individual papers in several previous conferences on computers in science; however, this appears to be the first time that the subject has been singled out for special emphasis. Certainly the role played by computers in spectral analysis is unique, and one observes that there is an ever increasing sophistication in their use.

The papers presented in this session are drawn from several different areas of research, and serve to point out some of the important applications of computers in curve fitting and band shape analysis. The methods are of course applicable to many other specific problems. For example, γ-ray and infrared spectroscopy are not formally represented although there is also considerable activity in spectrum analysis in these fields as evidenced by the current literature. The analytical use of γ-ray spectroscopy is an example of an application where the required curve fitting involves an extensive library of standard spectra with the implied requirement of a large amount of readily accessible storage. In general, the analysis of a spectrum is accomplished at some time after it has been recorded, but one of the papers presented here will discuss a real-time analysis application.

Several different types of approach to spectral analysis may be noted in the work to be discussed. For example, in some applications it is possible to calculate a spectrum based on a theoretical model for the types of interaction known to contribute to the experimentally observed spectrum. Comparison with experiment may then reveal the relative importance of individual interactions or species, and grossly wrong structures may become immediately obvious. The following paper on NMR spectra falls in this category. Other observed spectra stem from processes that are not sufficiently understood theoretically to permit simulation or perhaps are highly dependent upon the experimental conditions under which they are obtained. In such instances it may be useful to resolve the complex band structure into a number of components based upon a particular mathematical function. This is the case with much optical spectra, and the use of a modified Lorentzian (Lorentzian - Gaussian) band contour, as is discussed by one of the present papers is at the same time experimentally useful and consistent with the predictions of a simple theoretical treatment. Computer programs to optimize the fit of such functions to experimental curves are available in many computer libraries. As noted in the papers on ESR line shape fitting, simulation is again based on the use of Lorentz or Lorentz-Gaussian curves.

In high resolution mass spectrometry, the computer is required to accurately deduce the center of a line and test the shape of an envelope for unresolved multiplets. Here the data points are both accurate and form a smooth curve. A similar application exists in deducing the maximum energies of various types of nuclear decay processes, but in this case the noise levels are usually so high that some data averaging is required prior to curve fitting.

In all these applications the interpretation of the experimental data is now intimately related to computer operations. However, the emphasis on curve fitting techniques is self-limiting. As was pointed out in a recent review of absorption band shapes in liquids, the ability to analyze certain types of spectra has already reached a degree of excellence that challenges the experimentalist to reexamine the physics of the measurement. Certainly this is a healthy outcome of the successful application of computers.

---

[1] R.P. Young and R.N. Jones, Chem Rev. 71, 219 (1971)

A SCIENTIST'S VIEW OF NMR SPECTRUM SIMULATION

by

Lawrence C. Snyder
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey 07974

Computer simulation is one of several tools to answer questions and make predictions about nuclear magnetic resonance, NMR, spectra. My area of interest and research activity is in the development of NMR in liquid crystal solvents as a means of precise molecular structure determination.[1] Computer simulations of NMR spectra permit me to decide when agreement of theoretical and experimental spectra has been attained; thus confirming an assumed molecular structure, or the adequacy of an assumed set of magnetic resonance interactions in less familiar contexts. Computer simulations of spectra also permit me to predict the adequacy of NMR in liquid crystals for potential future applications in structure determination.

The principles of NMR are fairly well established, but their application requires considerable art. A theoretical calculation of NMR spectra requires that the system of nuclei being studied must be defined as well as the Hamiltonian which specifies their important interactions with each other, the applied magnetic field, and other electromagnetic fields of their environment. An important feature of nuclear spin systems is that the number of states of a system increases very rapidly with the number of nuclei. For a system of N protons each having two states, the total number of states is 2^N. Thus for systems of over twelve protons, matrices describing the vector space of states grow beyond the capacity of modern computers and are extremely expensive to diagonalize.

The Hamiltonian used to describe NMR of molecules in liquid crystal solvents is rather simple

$$H = -\frac{1}{2\pi}\sum_i \gamma_i(1-\sigma_i)H_z I_{zi} + \sum_{i>j} J_{ij}(I_{xi}I_{xj} + I_{yi}I_{yj} + I_{zi}I_{zj}) + \sum_{i>j} D_{ij}(-1/2 I_{xi}I_{xj} - 1/2 I_{yi}I_{yj} + I_{zi}I_{zj}) \qquad (1)$$

The first term gives the interaction of the nuclei with the applied magnetic field H, with provision for diamagnetic shieldings in the parameters σ. The interactions of nuclei of a molecule with each other by way of the molecular electrons are described by the indirect spin-spin coupling parameters J_{ij}, with which you are familiar from NMR in isotropic phases. The direct magnetic dipole-dipole interaction parameters D_{ij} are required only for NMR in anisotropic media such as liquid crystal solvents. The direct magnetic dipole interaction D_{ij} between nuclei i and j of a molecule is inversely proportional to the cube of the distance between the nuclei. Its evaluation by spectrum analysis makes possible the determination of precise molecular geometry. It is usually the major interaction which provides the structure of NMR spectra taken in liquid crystal solvents.

The main application of computer simulated NMR spectra is facilitate comparisons of theoretical and experimental spectra. Theoretical spectra can be presented as tables of line positions and intensities. However, as the number of nuclei and their states increases the length of such tables increases rapidly and the meaning of the entries is obscured. Computer simulation of the spectra is made by preparing a picture of the spectrum which results from summing these many lines with proper relative intensity and an assumed individual line shape (Lorentzian). The simulation immediately makes it possible to distinguish the overall structure of the spectrum from the maze of individual line components; in other words the forest from the trees.

When one is dealing with new and unfamiliar systems, comparison of these computer simulations with experimental spectra makes it possible to decide whether the basic Hamiltonian chosen is adequate to describe the system. This use is illustrated in Figure 1 which shows early simulations of the proton NMR of benzene in a liquid crystal solvent. The upper computer simulation spectrum was made using only direct dipolar interactions corresponding to a rigid planar hexagonal geometry for benzene. The central spectrum is experimental. The lower simulated spectrum was computed using both the direct and indirect couplings of nuclei. These simulations showed that the main structure of NMR spectra of benzene in liquid crystal solvents is due to the direct dipolar couplings D_{ij} of the Hamiltonian.

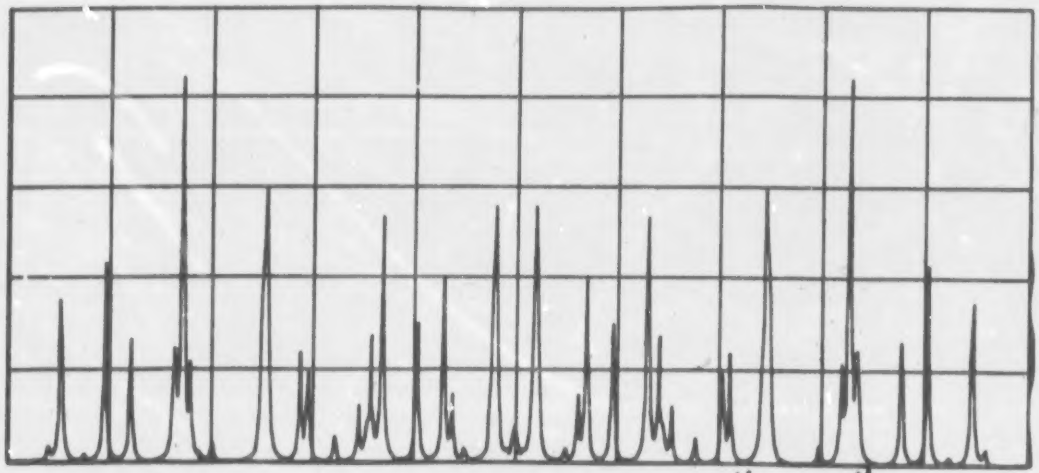
When the proper Hamiltonian is well established and one is dealing with familiar systems, computer simulations permit one to deduce structural information from NMR in anisotropic solvents. An idea of the amount of information available was given by the computer simulations shown in Figure 2 for benzene containing a single carbon-13 nucleus. The C-H bond lengths assumed differ by only 0.005 Å but produce clear visual differences in the computer simulated spectra. Similarly, it was easily shown that the NMR in liquid crystal would clearly distinguish benzene from possible isomers having a three-fold or greater axis of symmetry.

The use of computer simulation in an interactive mode at a Graphic-1 console proved to be a convenient way to determine a small number of structural parameters when the NMR spectra are relatively simple as for cyclopropane. However, when the number of parameters which must be varied is greater, the interactive mode of analysis is less satisfactory, particularly when the number of nuclei is larger and the NMR spectra correspondingly more complex as is illustrated by bicyclobutane. Small changes of assumed geometry for bicyclobutane caused the lines of this complicated spectrum to move through one another in ways which defied identification and analysis. The Graphic console was an inadequate tool and analysis was completed by provision of a different experiment, double resonance. The computer simulation in Figure 3 of the proton NMR of bycyclobutane corresponding to that analysis showed definitively its validity.

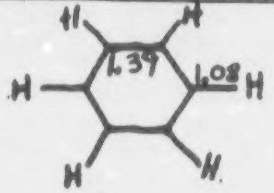
The problems of computing and analyzing NMR spectra of molecules having large number of nuclei are met in cyclohexane,[2] which has twelve protons. The NMR spectrum in a liquid crystal is symmetric and consists of two gaussian shaped collections of resonance lines. For cyclohexane, the matrices involved are of dimension 1000x1000 and too large for convenient and economical diagonalization. Even after a costly computer simulations, if possible, one could not extract the interesting direct dipolar interactions by fitting these rather amorphous masses of resonance. Again the judicious way to proceed has turned out to be an improvement of the experiment; partial deuteration of the cyclohexane with decoupling of the deuterons by double resonance. Here again, computer simulations, are providing evidence on the adequacy of our understanding of double resonance in this new context of anisotropic solvents.

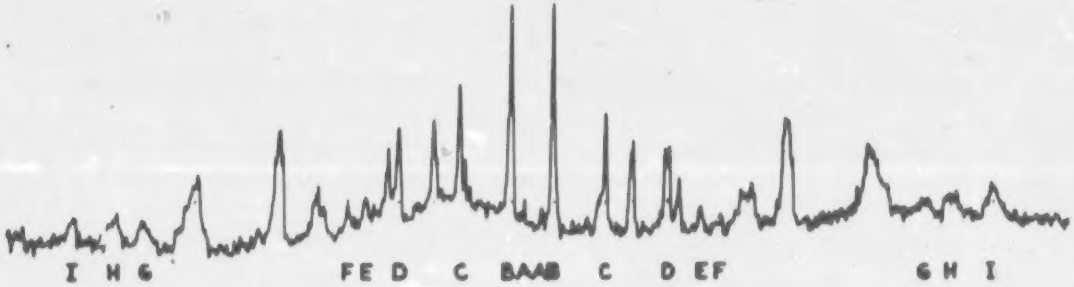
In summary, computer simulation is another tool to answer interesting questions in magnetic resonance. It is taking its place among the alternative experimental and theoretical tools.

1. L. C. Snyder, J. Chem. Phys., 43, 4041 (1965).
2. S. Meiboom and L. C. Snyder, Accts. of Chem. Res., 4, 81 (1971).

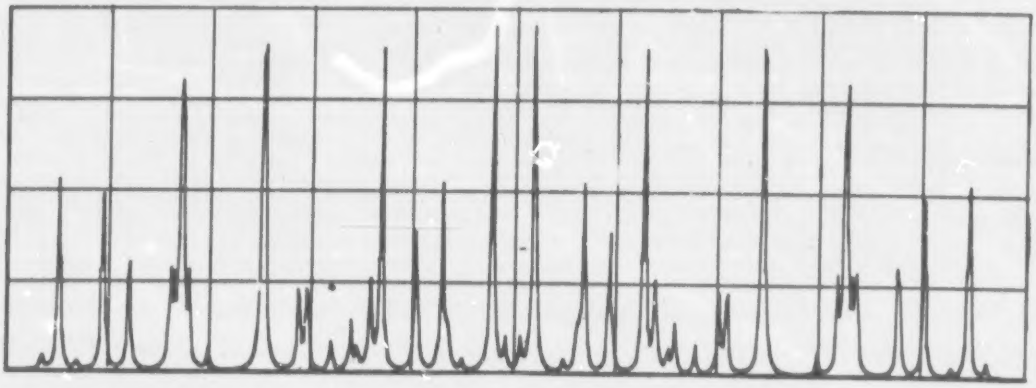


Hexagonal Planar Benzene
Dipolar Coupling Only



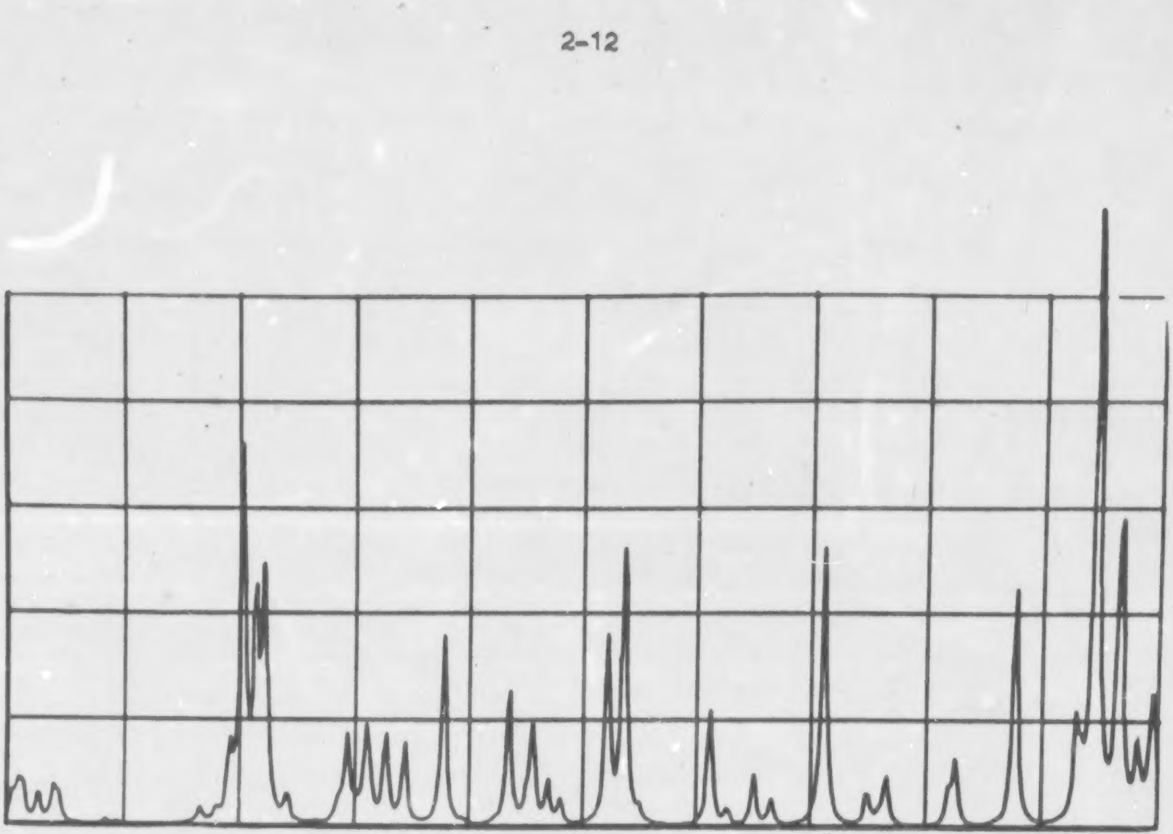


Exp Benzene Proton NMR In Liquid Crystal Solvent

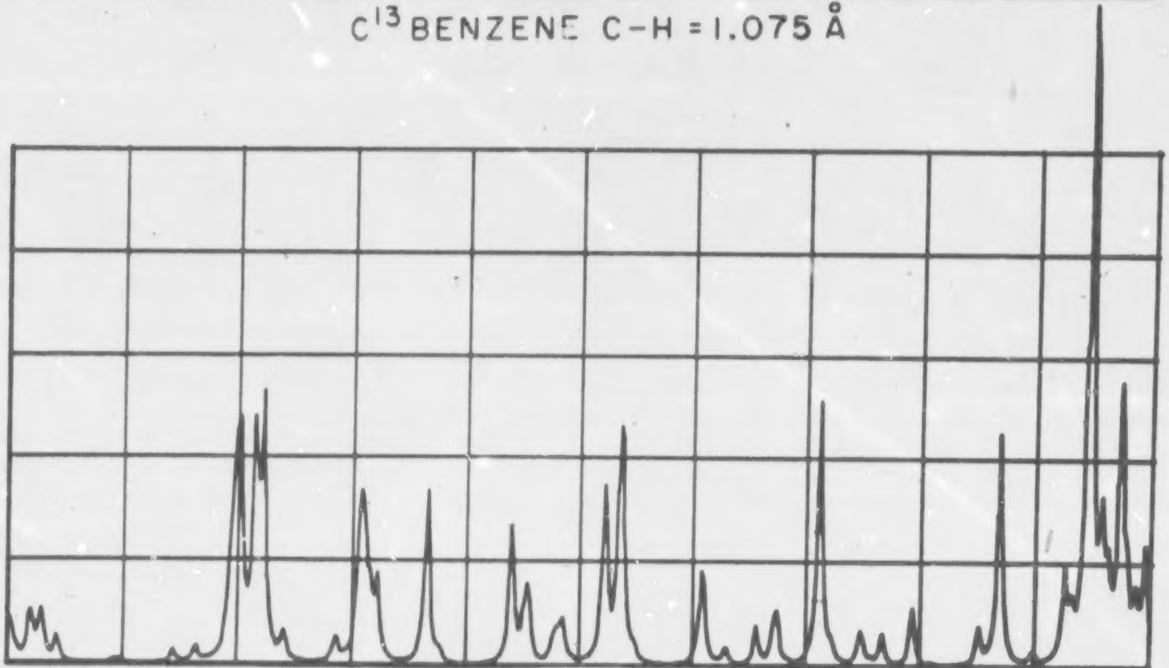


Hexagonal Planar Benzene
Dipolar and Indirect Coupling

Figure 1



C^{13} BENZENE C-H = 1.075 Å



C^{13} BENZENE C-H = 1.08 Å

Figure 2

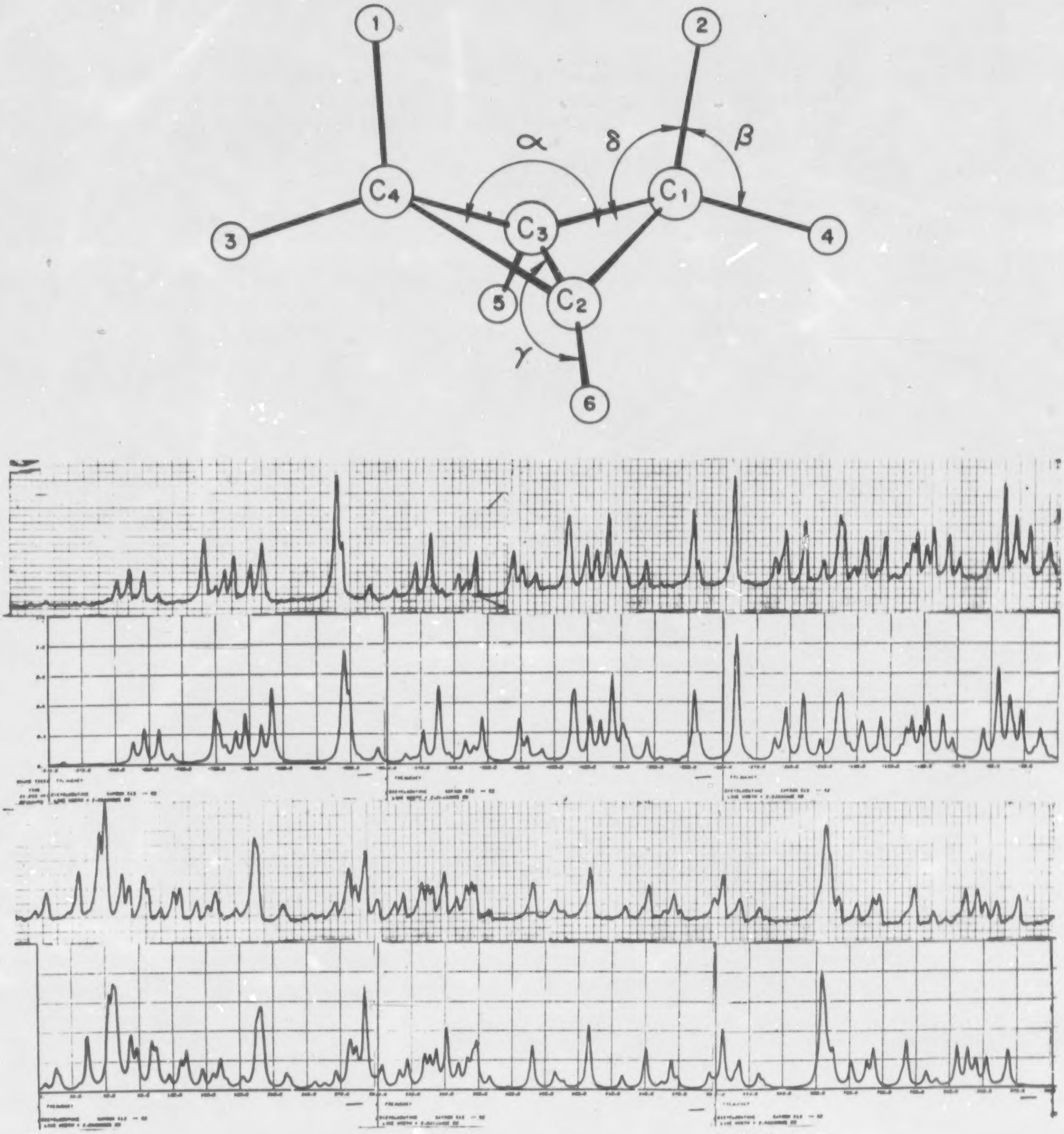


Figure 3 - Bicyclobutane Proton NMR: Experimental/Computer Simulation.

AN ESR LINESHAPE ANALYSIS FOR DETERMINATION OF UNRESOLVED METAL HYPERFINE SPLITTINGS IN ION PAIRS: ITS APPLICATION TO THE BENZENE ANION RADICAL*

M. T. Jones, M. Komarynsky, and R. D. Rataiczak[1]

Contribution from the Department of Chemistry
University of Missouri-St. Louis
St. Louis, Missouri 63121

ABSTRACT

A lineshape analysis is presented which allows the extension of the study of ion-pairing into the spectral region where the metal hyperfine splitting (hfs) can no longer be resolved. The analysis allows one to quantitatively evaluate the metal hfs and the component linewidths from experimentally measured lineshapes and linewidths. The utility of the analysis procedure is demonstrated by its application to the benzene anion radical. The unresolved metal hfs in the benzene anion radical ranges from -75 milligauss at -120°C to +150 milligauss at -30°C. The component linewidths are found to range from ~200 milligauss at -120°C to ~650 milligauss at -30°C.

INTRODUCTION

There has been considerable interest in the use of ESR techniques to study ion-pairing effects.[2] To date, it has been necessary to explicitly resolve the metal hyperfine splitting (hfs) in order to study such effects. Although, very recently certain changes in g-values have also been associated with ion-pairing.[3] A lineshape analysis is presented which will allow the extension of the study of ion-pairing into the spectral region where the metal hfs can no longer be resolved. The analysis allows one to quantitatively evaluate the metal hfs and the component linewidth (hence to obtain the "true" T_2) from experimentally measured lineshapes and linewidths. The utility of the analysis procedure will be demonstrated by its application to the benzene anion radical.[4]

The underlying basis of the analysis procedure is the fact that ESR spectral lineshapes are strongly dependent upon the metal hfs even when the individual components are not resolved. The analysis consists of determining precisely what the lineshape looks like as the metal hfs increases from zero to the point where the individual components just begin to become resolved. For the calculations described here, it is assumed that the life time of the cation-anion pair is long compared to the inverse of the metal hfs.

EXPERIMENTAL AND RESULTS

The spectral line shapes were simulated for an ESR line width consists of n-closely spaced individual component lines. For example, in the case of ion-pair formation with a single sodium ion, four equally intense component lines are used. The simulation calculations were performed in such a manner that errors due to truncation of the individual component lines fell outside of the region of interest. The density of points calculated was such that the percentage of error associated with picking the correct half-width at extreme slope ranged from $\pm 1\%$ when $(Hfs/\Delta H_c)=0$ to $\pm 0.25\%$ when $(Hfs/\Delta H_c)=1.0$ and $\pm 0.5\%$ when $(Hfs/\Delta H_c)=0$ to $\pm 0.3\%$ when $(Hfs/\Delta H_c)=1.6$ for Lorentzian and Gaussian component lineshapes, respectively. The component linewidth is given by ΔH_c. Figure I shows normalized plots of half of the derivative of the resonance linewhape for various ratios of hfs to ΔH_C starting with four equally intense Lorentzian shaped components. The data in Figure I show that as the hfs increases from zero relative to ΔH_C the lineshape becomes less and less Lorentzian. It falls off in the wings much more rapidly than a Lorentzian. For hfs:ΔH_C ratios >0.6 the lineshape falls off even more rapidly than a single Gaussian component. Similar behavior is observed when Gaussian components are used except that their limiting lineshape is a Gaussian and they fall off more rapidly than a Gaussian in the wings as the ratio of hfs:ΔH_C is increased.

Figure II shows a plot of the ratio of the calculated linewidth (ΔH_{Obs}) to the individual component line width (ΔH_C) versus the ratio of Hfs:ΔH_C. One notes that as the ratio of Hfs:ΔH_C is increased so too does ΔH_{Obs}. A similar result is observed when Gaussian components are used. The results shown in Figures I and II are changed by an insignificant amount if one choses instead of four equally intense lines, two sets of four equally intense lines with relative intensity and relative hfs appropriate to the case of ion-pair formation with potassium ions instead of sodium.

The question immediately arises: Are there examples of the behavior of lineshapes as described above? The answer, of course, is yes. Past studies on very dilute water solutions (9.5×10^{-5} M) of peroxylamine disulfonate anion radical have shown that the lineshape is non-Lorentzian with a more rapid fall off in the wings.[8] Figure 4 of reference 8 shows a normalized plot of the derivative of the resonance absorption curve of peroxylamine disulfonate anion radical in comparison with pure Lorentzian and Gaussian line shapes.[9]

Careful studies of the lineshape of the benzene anion radical in solution show that, a) it is dependent upon the benzene concentration, b) it is temperature dependent, and c) it falls off more rapdily in the wings than does a Lorentzian line.[6,7,10] This behavior is demonstrated in Figure III for a solution of the benzene anion radical in a 2:1 mixture of tetrahydrofuran and 1,2-dimethoxyethane which is 0.3M in benzene.

Figure III shows another way of looking at the information shown in Figure I. On the ordinate is plotted the "observed" line position versus that of a pure Lorentzian line for fixed percentages of the height. The analysis procedure consists of marking off the baseline in units of one-half the peak-to-peak width. The height of the curve is marked off in percentage of the maximum height (which occurs at ±1/2 the peak-to-peak width) from the base line. The distance (normalized with respect to one-half the peak-to-peak width) from the center of the resonance line to the decaying wing at a given percentage of the height is measured and plotted as shown in Figure III against the corresponding value for a pure Lorentzian line. A pure Lorentzian lineshape yields points which fall along the dashed line.

Whereas the data plotted in Figure III are "experimental", the data shown in Figures IV and V are obtained from calculations of the expected lineshape of the central line in the benzene anion radical excluding and including the effect of the hfs arising from the natural abundance of ^{13}C, respectively.[11] Note that in Figure IV there are a series of straight lines coming out of the origin. The larger the ratio metal Hfs:ΔH_C the lower the slope. The inclusion of the effect of the ^{13}C hfs leaves a portion of the data in Figure IV unchanged, namely that portion along the abscissa from 1 to approximately 3. For distances greater than 3 halfwidths, the individual curves are pulled down toward the abscissa as shown in Figure V. Note the good agreement between the data shown in Figure III and that in Figure V.

In Figure VI is shown a plot of the data obtained from our calculations which allows a rather easy comparison of the calculations with experimental results. Line positions in units of half-widths are plotted versus the ratio Hfs:ΔH_C for a series or family of curves taken at different relative heights. Our calculations show that the positions associated the points at 80%, 60%, and 40% of the maximum height are not significantly perturbed by the presence of the ^{13}C hfs.[13] Thus, we may safely compare "experimental" positions at 80%, 60% and 40% of the maximum height with the curves shown in Figure VI. One obtains in this way the "best" fit of the experimental data and the value of the ratio Hfs:ΔH_c at which it occurs. Using Figure II, one can convert this ratio into a number which gives the ratio of the "observed" linewidth to the component linewidth. These two ratios plus the experimentally measured linewidth yield the information we are seeking, namely, the metal hfs and the component linewidth.

Figure VII shows the results of such an analysis for the metal hfs for a large range of benzene anion radical concentrations and temperatures. Of course, the analysis is not capable of determining the sign of the hfs, although we have plotted the data through the origin. There is no incontrovertible proof that the metal hfs does in fact change sign. However, one can argue in analogy with a number of other ion-paired radicals where the hfs values are known to go through zero.[2a,2c,2f] It is significant that the metal hfs values are relatively independent of the benzene concentration and hence the benzene anion radical concentration. This fact taken together with the fact that the method of analysis appears to work over such a large range of benzene and benzene anion concentrations suggests that the assumption, that the life-time of the benzene anion radical-metal ion pair is long compared to inverse of the metal hfs, is justified.[14]

Figure VIII shows a plot of the component linewidth versus temperature for the same range of concentration as shown in Figure VII. The linewidths are dependent upon concentration, although the scatter is larger than one might like to see. In addition, the temperature dependence appears to level off at about -80°C. This is to be contrasted with the inhomogeneously broadened line which displays a minimum at -100°C.[10]

ACKNOWLEDGEMENT

The partial support of this research by the National Science Foundation through grant GP-15619 is acknowledged. The support of the University of Missouri-St. Louis Computer Center is also acknowledged as is the support of RDR by NASA through grant #NsG(T)-74.

REFERENCES

*Presented at the Second Symposium on Electron Spin Resonance held at the University of Georgia, Athens, Georgia, December 7-9, 1970.

1. Present address: Incarnate Word College, San Antonio, Texas 78209.

2. See for example a). E. de Boer, Recl. Trav. Chim. Pays-Bas 84, 609 (1965); b). N. M. Atherton and S. I. Weissman, J. Am. Chem. Soc. 82, 2537 (1960); c). N. Hirota, J. Am. Chem. Soc. 90, 3603 (1968); d) M. T. Jones and D. E. Zeller, J. Chem. Phys. 48, 334 (1968); e). N. Hirota, Chapter 2, in "Radical Ions" edited by E. T. Kaiser and L. Kevan, Interscience (1968); f). I. B. Goldberg and J. R. Bolton, J. Phys. Chem. 74, 1965 (1970).

3. C. L. Dodson and A. H. Reddock, J. Chem. Phys, 48, 3226 (1968); b). W. G. Williams, R. J. Pritchett, and G. K. Fraenkel, J. Chem. Phys. 52, 5584 (1970).

4. The suggestion that the benzene anion radical might be ion-paired is not original with the work reported here. However, the evidence presented leaves no doubt that the benzene anion radical in solution is ion-paired. Malinoski and Bruning[5] first suggested that the benzene anion radical might be ion-paired on the basis of their studies of the rate of electron transfer between the benzene anion radical and neutral benzene and their attempts to simulate the electron transfer broadened ESR spectra. Their suggestion was criticized by Kooser, Volland, and Freed[6] (See their footnote 52a.) who argued among other things that published simulated spectra [cf., Y. S. Lebedev et al, "Atlas of Electron Spin Resonance Spectra", (Consultants Bureau, New York, 1963)] decayed more slowly in the wings than did a pure Lorentzian lineshape which was contrary to experiment. Subsequently, Das, Wagner, and Freed[7] on the basis of more experimental evidence revised their criticism of Malinoski and Bruning's suggestion. Apparently, their revision includes a reinterpretation of the simulated spectra of Lebedev et al which now agrees qualitatively with the results reported in this paper.

5. G. L. Malinoski and W. H. Bruning, Angew. Chem. Internat. Ed. 7, 953 (1968); J. Chem. Phys. 50, 363 (1969).

6. R. G. Kooser, W. V. Volland, and J. H. Freed, J. Chem. Phys. 50, 5243 (1969).

7. M. R. Das, S. B. Wanger, and J. H. Freed, J. Chem. Phys. 52, 5404 (1970).

8. M. T. Jones, J. Chem. Phys. 38, 2892 (1963).

9. While the analysis of the peroxylamine disulfonate spectra lineshapes will not be considered further here, it is of interest and studies in our laboratory are underway.

10. a) R. D. Rataiczak, Ph.D. Thesis, St. Louis University (1970); b) R. D. Rataiczak and M. T. Jones, J. Chem. Phys. (submitted).

11. J. R. Bolton, Mol. Phys. 6, 219 (1963) reports that the value of the ^{13}C hfs is 2.8 (±0.1) gauss at -100°C. Additional work in our laboratory shows that it is temperature dependent and that it increases with increasing temperature. This latter observation agrees with that of Danner and Maki[12] for the hexakis (trifluoromethyl) benzene anion radical. Our values are 2.70 gauss at -100°C, 2.79 gauss at -80°C, and 2.85 gauss at -62°C.

12. J. C. Danner and A. H. Maki, J. Am. Chem. Soc., 88, 4297 (1966).

13. Our calculations show that for ^{13}C Hfs:ΔH_c ratios greater than 5 and metal Hfs:ΔH_c ratios less than approximately 0.36 (as the former becomes larger so does the latter ratio) that the presence of the ^{13}C Hfs can be ignored in the analysis procedure. Very few spectra have been observed to date which do not fall into the above category.

14. This observation is consistent with the low electron transfer rate between the benzene and the benzene anion radical observed by Malinosky and Bruning (Ref. 5) and suggests that the electron transfer takes place with the simultaneous transfer of the metal ion (see Ref. 10b for an extended discussion of this point). We plan further comput-

er simulations which will include the effects of metal ion exchange and electron transfer. However, we are, at present, quite impressed with the success we have had to date by neglecting such effects and the lack of evidence that they need to be included for adequate spectra simulation.

CAPTIONS FOR FIGURES

FIGURE I. Normalized plot of half of the derivative of the resonance lineshape for various ratios of hfs to ΔH_C. The individual components consisted of four equally intense Lorentzian shaped lines. The values of the ratio hfs:ΔH_C shown are 0, 0.17, 0.34, 0.52, and 0.69 in order of decreasing amplitude in the wings. The dotted line represents a pure Gaussian lineshape.

FIGURE II. Plot of the ratio of the calculated linewidth (ΔH_{obs}) to the individual component line width (ΔH_C) versus the ratio of hfs:ΔH_C. The individual components consisted of four equally intense Lorentzian shaped lines. When the ratio hfs:$\Delta H_T \geq 0.86$ the individual component lines just begin to be resolved.

FIGURE III. Lineshape of the $M_H=0$ component in the first deviative ESR spectrum of the benzene anion radical in 2:1 (THF-DME) solvent (0.3M benzene) at different temperatures. The dashed line traces the locus of the Lorentzian line.

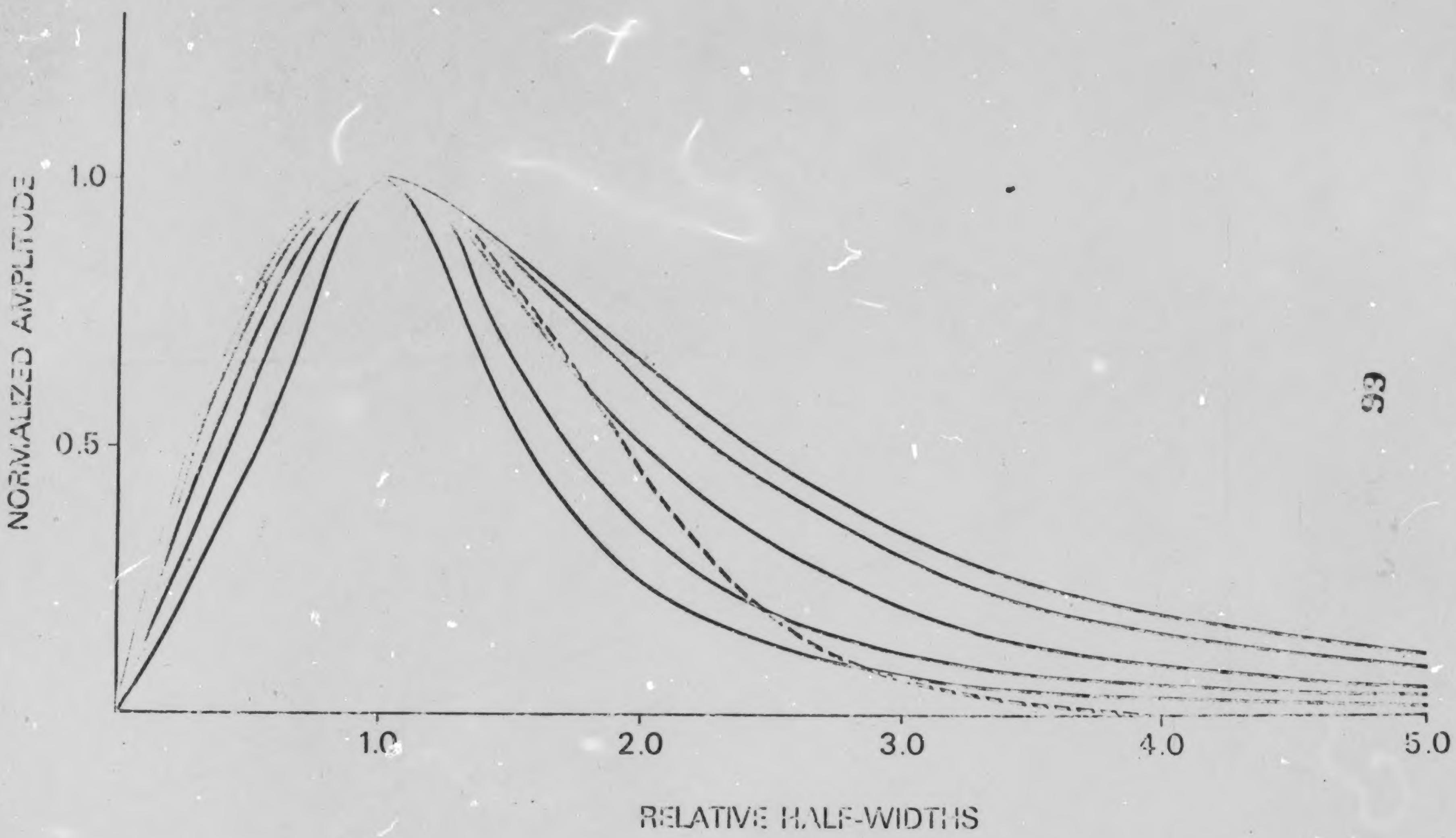
FIGURE IV. Plot of the "observed" or "calculated" resonance line position versus the "theoretical" Lorentzian position for a series of increasing values of the metal Hfs:ΔH_C ratio. The effect of ^{13}C Hfs is not included.

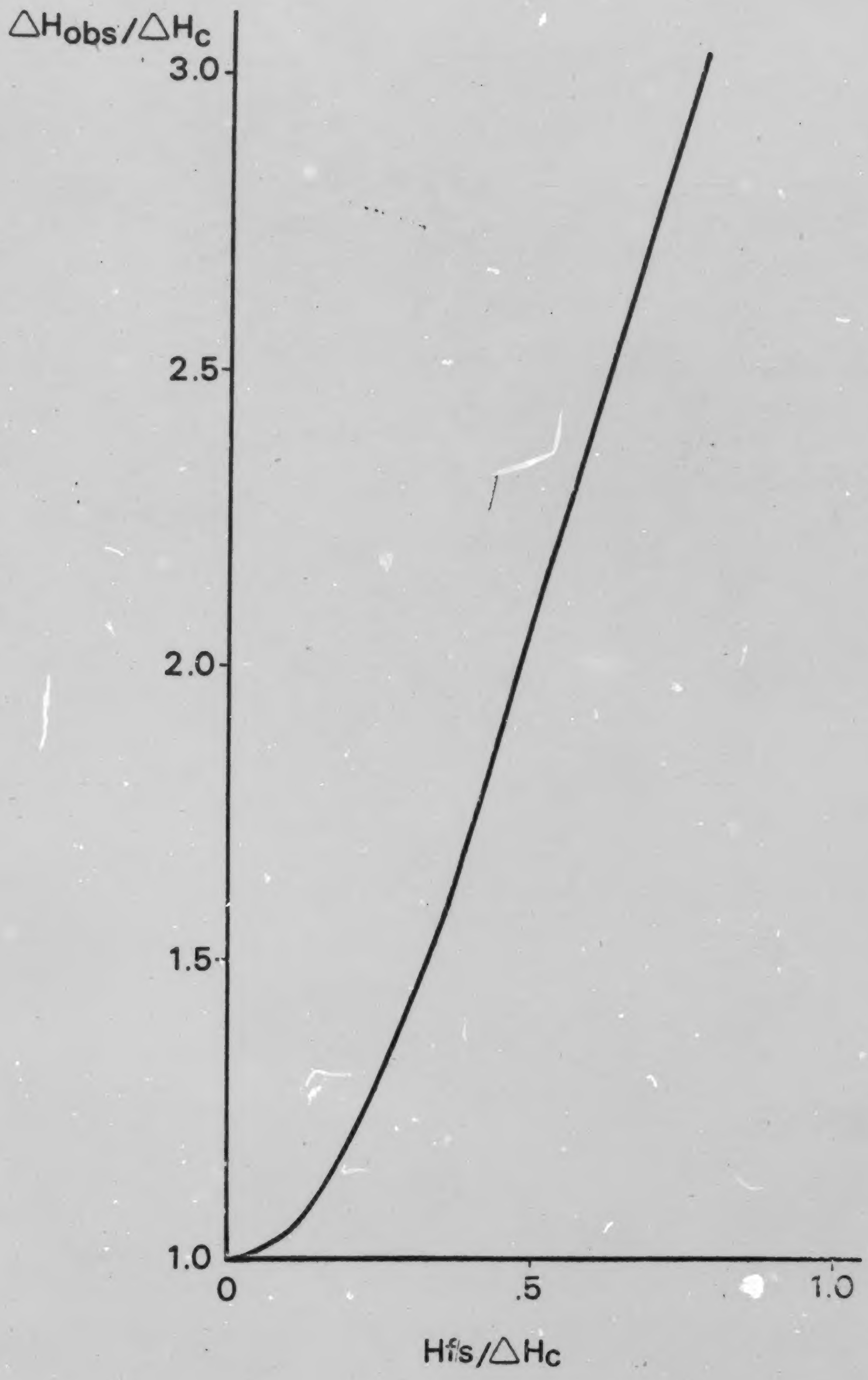
FIGURE V. Plot of the "observed" or "calculated" resonance line position versus the "theoretical" Lorentzian position for a series of increasing values of the metal Hfs:ΔH_C ratio. The ^{13}C Hfs:ΔH_C ratio for the calculations shown here is 8.

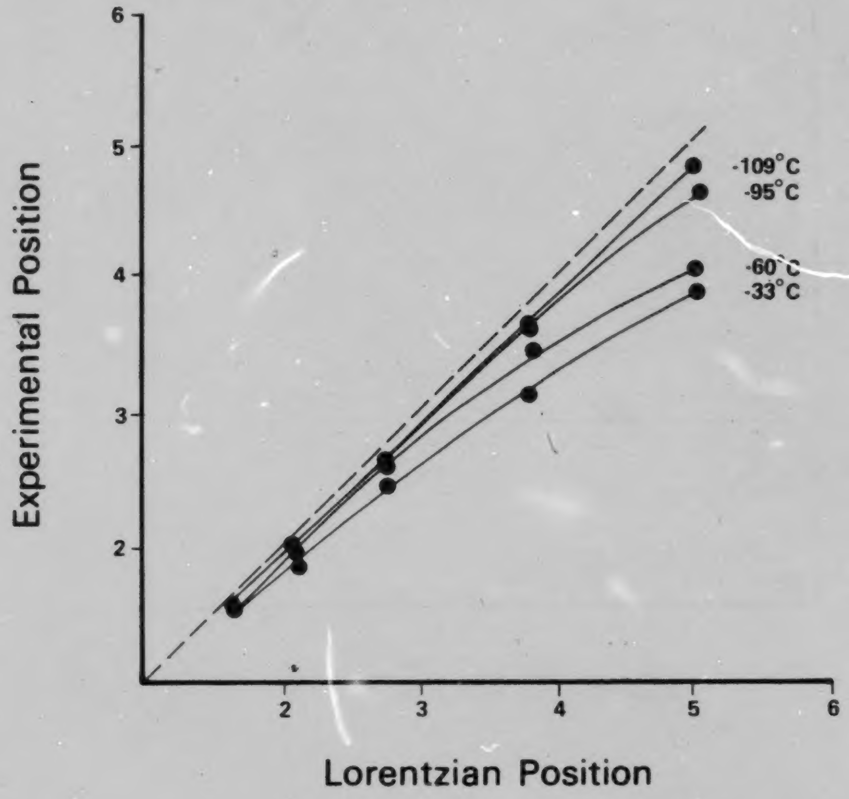
FIGURE VI. Plot of line positions in units of peak-to-peak half width versus the metal Hfs:ΔH_C ratio for various different relative heights. The effect of ^{13}C Hfs is not included in this plot.

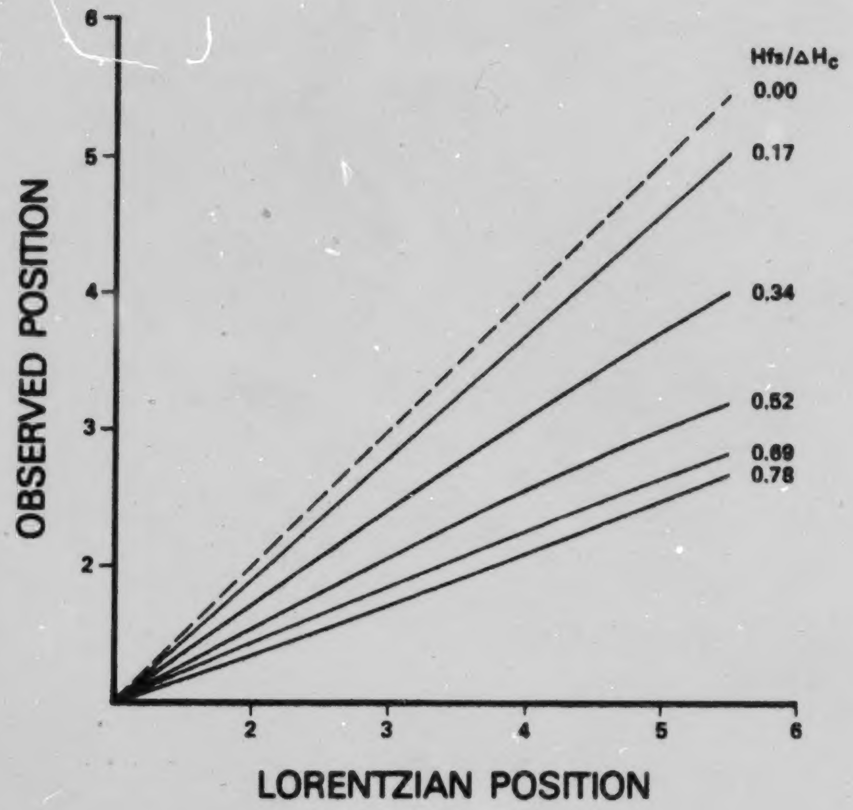
FIGURE VII. Plot of the metal hfs versus temperature for a series of different benzene concentrations.

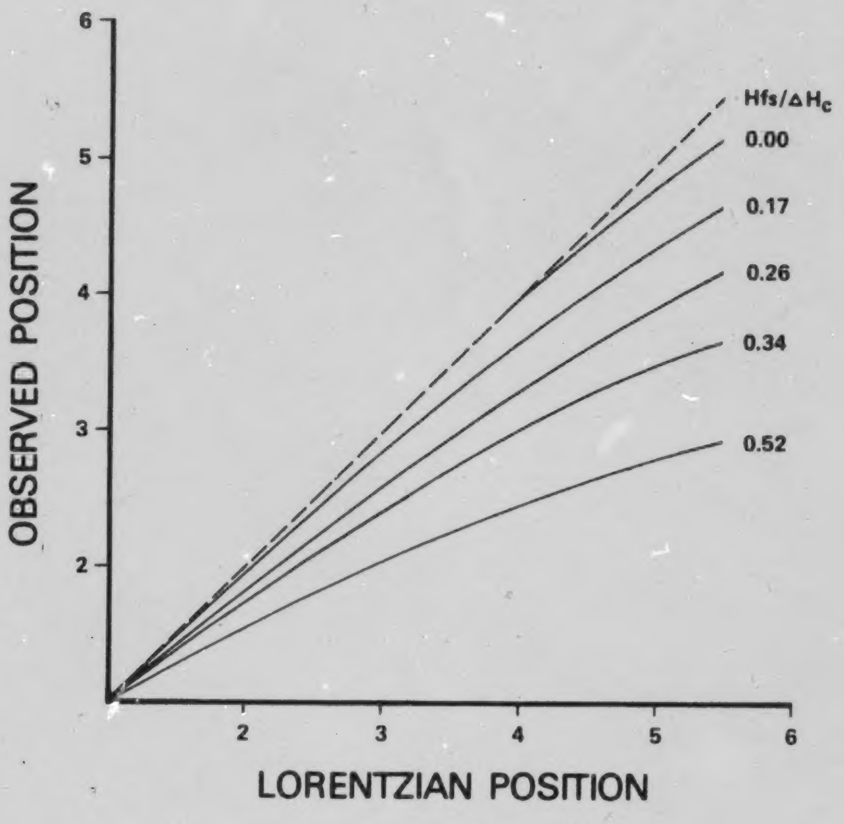
FIGURE VIII. Plot of component linewidth (ΔH_C) versus temperature for a series of different benzene concentrations.

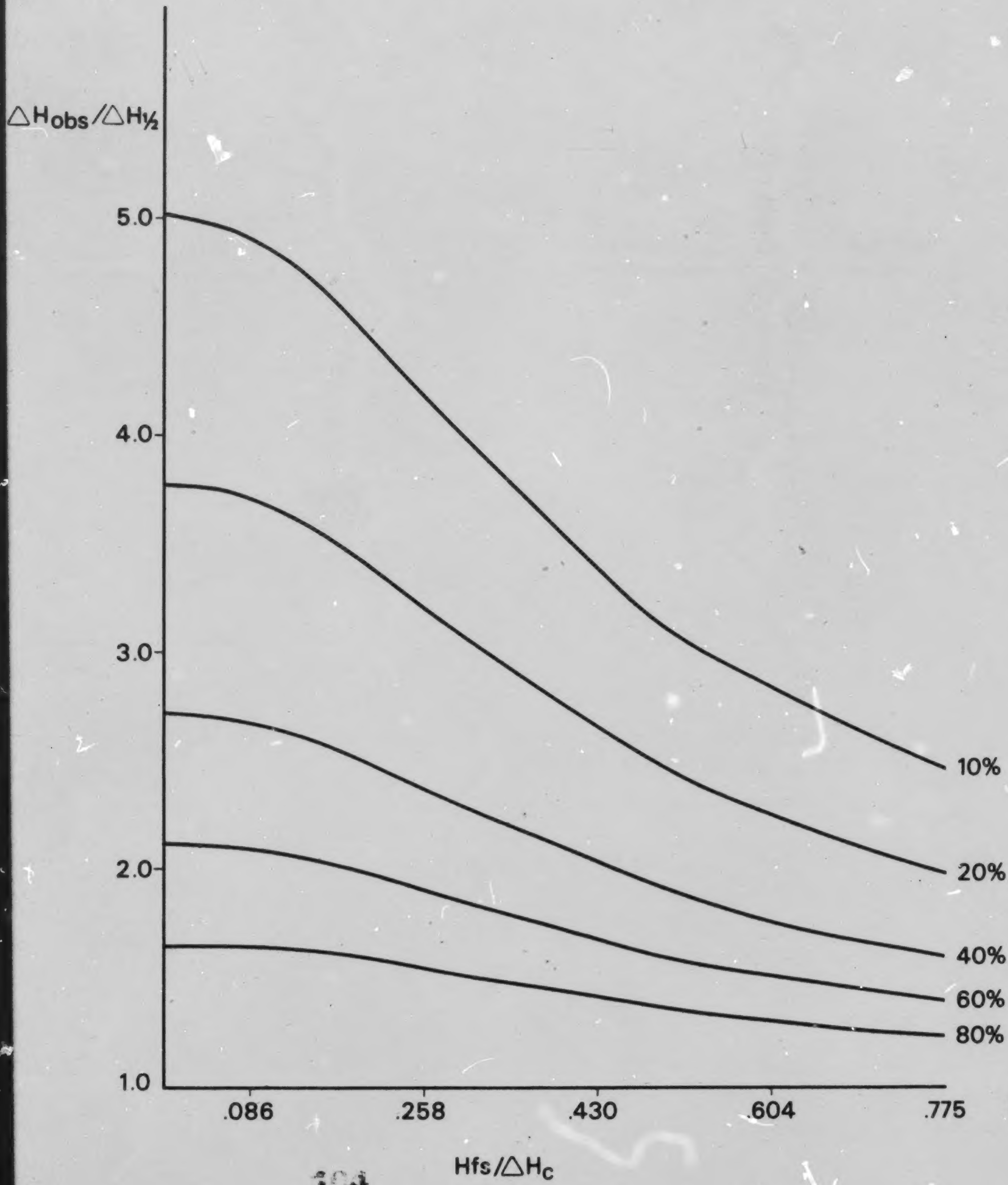

NORMALIZED AMPLITUDE
1.0
0.5
1.0
2.0
3.0
4.0
5.0
RELATIVE HALF-WIDTHS


$\Delta H_{obs}/\Delta H_c$
3.0
2.5
2.0
1.5
1.0
0
.5
1.0
$H_{fs}/\Delta H_c$

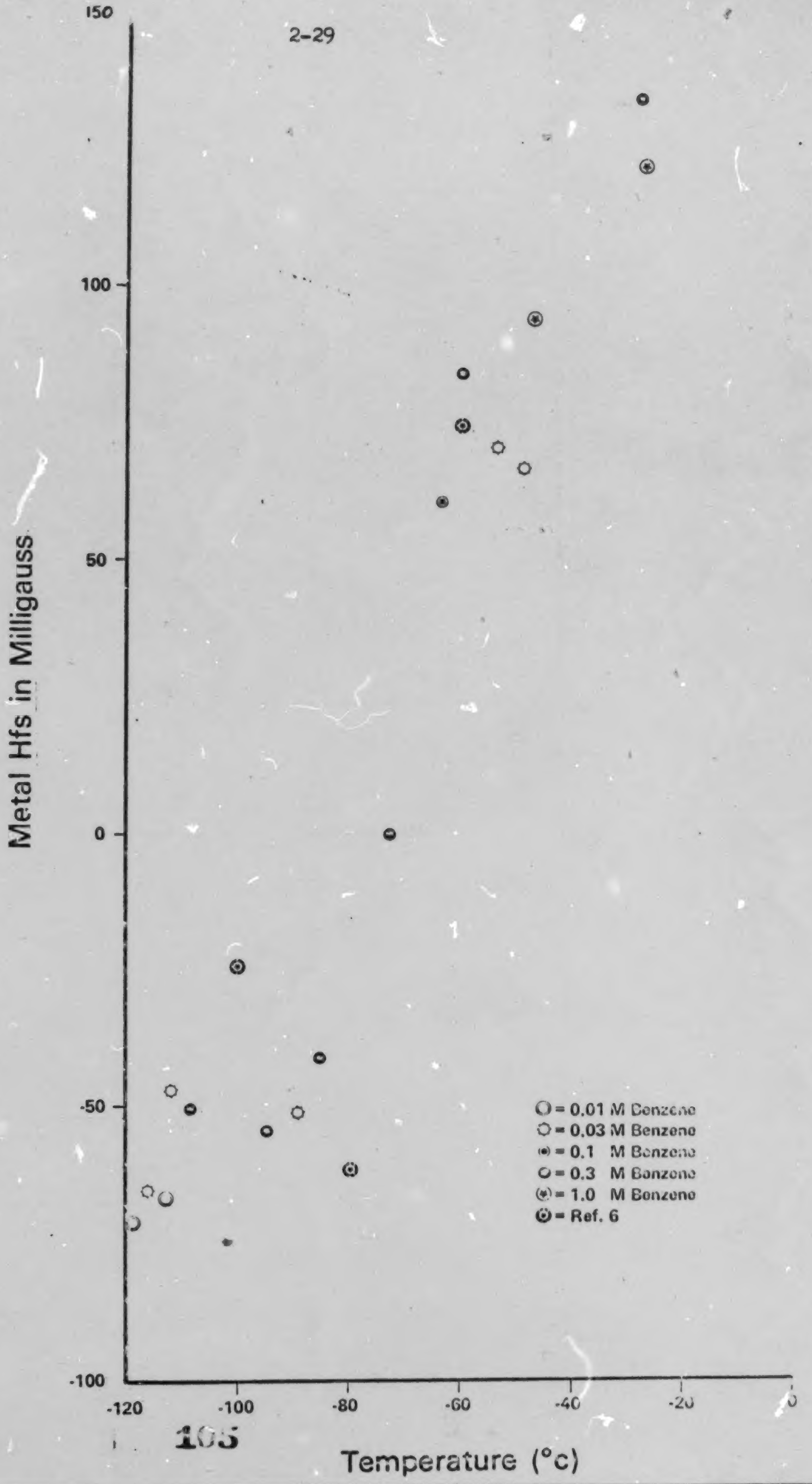

Experimental Position
Lorentzian Position
6
5
4
3
2
2
3
4
5
6
-109°C
-95°C
-60°C
-33°C

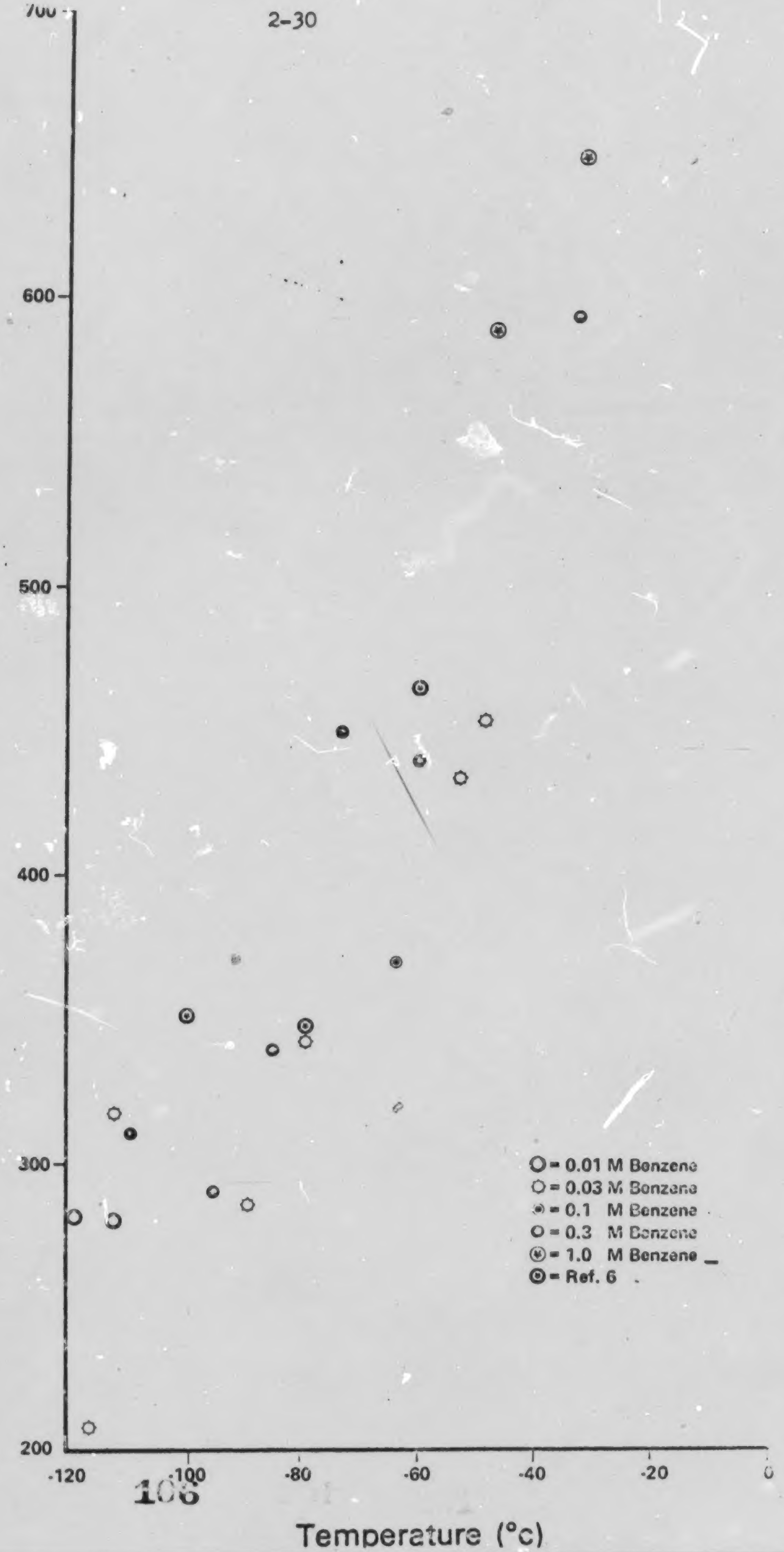

OBSERVED POSITION
6
5
4
3
2
LORENTZIAN POSITION
2
3
4
5
6
Hfs/ΔH_c
0.00
0.17
0.34
0.52
0.69
0.78


6
5
4
3
2
OBSERVED POSITION
2
3
4
5
6
LORENTZIAN POSITION
Hfs/ΔHc
0.00
0.17
0.26
0.34
0.52


$\Delta H_{obs}/\Delta H_{1/2}$
5.0
4.0
3.0
2.0
1.0
.086
.258
.430
.604
.775
$H_{fs}/\Delta H_c$
10%
20%
40%
60%
80%

2-29


150
100
50
0
-50
-100
Metal Hfs in Milligauss
-120
-100
-80
-60
-40
-20
0
Temperature (°c)
○ = 0.01 M Benzene
○ = 0.03 M Benzene
● = 0.1 M Benzene
○ = 0.3 M Benzene
● = 1.0 M Benzene
● = Ref. 6


700
600
500
400
300
200
-120
-100
-80
-60
-40
-20
0
Temperature (°c)
○ = 0.01 M Benzene
○ = 0.03 M Benzene
● = 0.1 M Benzene
○ = 0.3 M Benzene
⊛ = 1.0 M Benzene
⊙ = Ref. 6

2-31

Evaluation of Complex E.S.R. Spectra by Computer

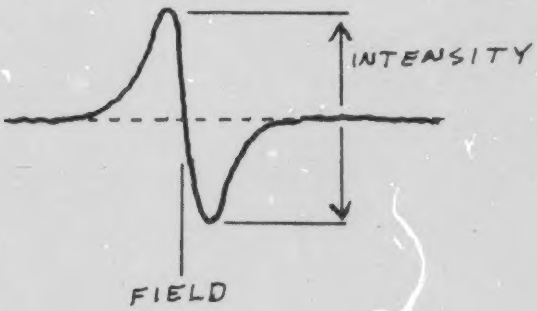
William E. Bennett, Robert E. Buckles
Department of Chemistry
University of Iowa
Iowa City, Iowa

Introduction

The interpretation of complex E.S.R. spectra can prove difficult, even with the computer, when the number of theoretical lines exceeds a few hundred. The primary difficulty is the fact that in such cases there is considerable accidental overlap of lines. The authors have found that one can resolve such spectra quite easily by using simplifications which considerably speed the computation. If the computation is sufficiently fast, one may try hundreds or thousands of splitting constants in a reasonable length of time. This paper is a report of the results of this approach.

Method

The simplest description of an E.S.R. line is by its height and by its field position relative to the center line.



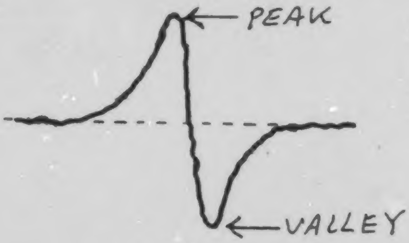
If we have an exact accidental overlap of two lines, the resulting peak will have a height which is the sum of the two. This is true if both lines have the same width, which is generally the case for organic radicals. If the overlap is not exact, this is not true. The authors have found, however, that one can rapidly eliminate all but a few of the possible combinations of splitting constants by assuming that overlapping lines sum their intensities. Another approximation is that neighboring peaks do not affect each other.

The experimental line intensities and field positions are introduced as data. Guesses of splitting constants or of incrementing information are then supplied.

The program computes the theoretical line spectrum, summing lines which fall under measured peaks. If no computed lines fall near an experimental peak, the splitting constants are rejected. A standard deviation for the fit between the theoretical and experimental spectrum is computed. This is given in the units of the experimental spectrum (usually cm or mm.). If the standard deviation is less than a value supplied with the data, a print-out of the experimental and computed spectral line intensities is given.

This first phase of the computation narrows the number of possible combination of splitting constants from several hundred to a few.

The second phase of the calculation is to introduce the splitting constants, which were selected by phase one, as data, along with an estimate of the line width. This phase computes a theoretical spectrum assuming a Lorenzian line shape. Time is saved in that the intensity need be computed only at the peak and valley of the theoretical line, which may be a composite of many lines.



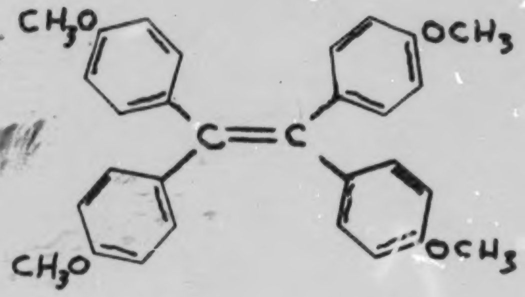
To conserve time, the line shape function is in the form of a table.

A standard deviation is again computed. This second step usually narrows the selection of splitting constants to one or two possibilities.

A final third phase is to refine the line width to give the lowest standard deviation. The best splitting constants from phase two are used. This final phase generally narrows the choice to one set of constants. Plotting the computed spectrum verifies this further.

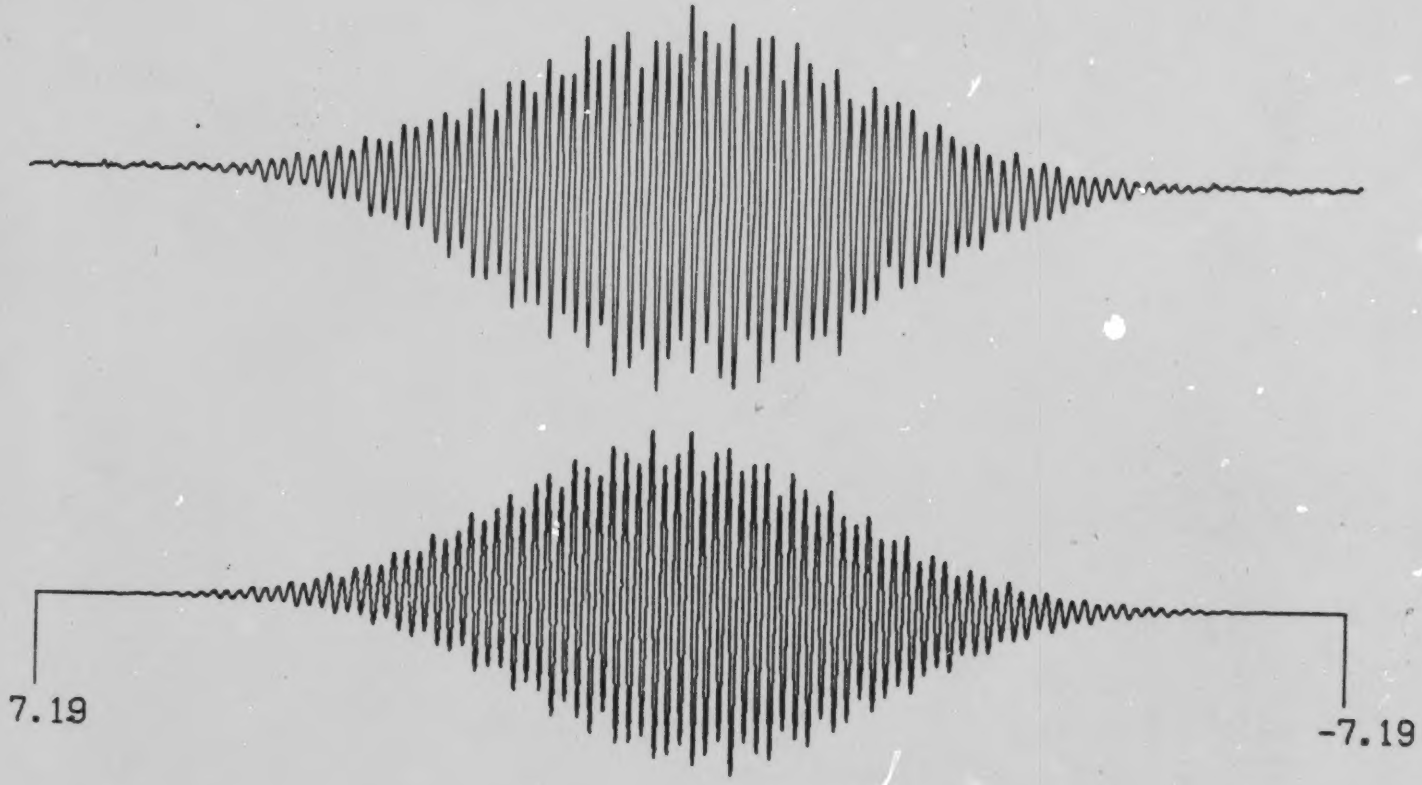
Results

Tetraanisylethylene forms a radical cation which has



1053 theoretical lines. The observed spectrum is given in Figure 1 and has 89 lines. The regular spacing of the lines indicates that the three splitting constants have a common factor or nearly so. The spacing is about 0.141 gauss. The splitting constant can then be found by incrementing by the spacing.

The first phase of the computation gave, out of many hundred increments, several possible sets of splitting constants with low standard deviations. The second phase narrowed this to two. The third phase gave one. The computed spectrum is given in Figure 1. The splitting constants were: 8H = 0.705 gauss, 8H = 0.420 gauss and 12H = 0.844 gauss. The refined width was 0.078 gauss.



$(\mathrm{TAE})^{+\cdot}$

SPLITTINGS 0.705. 0.420. 0.844---WIDTH 0.078

FIG. 1

The results were further confirmed by fitting the spectrum obtained when the ethylene carbons were enriched with carbon 13.

An additional program which the authors have used subsequent to the above work, preprocesses the experimental data for phase one. For this program a line width is assumed. An approximation is made of the intensities the experimental lines would have with very narrow width. This is, in effect, a resolution enhancement and gives a more exact fit in phase one. This conditioned spectrum is not used in phase two or three.

Structure Determination of Organic Molecules by Mass Spectrometry

K. Biemann
Department of Chemistry
Massachusetts Institute of Technology
Cambridge, Ma. 02139

The vast amount of data which a modern mass spectrometer can produce within a short time has, over the past few years, forced the chemist to rely more and more on computers to ensure the full and efficient exploitation of these data. The techniques fall into three general categories: The high speed by which mass spectra are recorded in certain situations requires on-line data acquisition; the large volume of raw data (mainly time-intensity pairs) reprints fast and efficient processing (to mass-intensity pairs, and, in the case of high resolution data, further to elemental composition-abundance pairs); and finally interpretative methods to identify the compound giving rise to the spectrum or to aid the chemist in this interpretation.

Historically, the first instance of at least semi-automatic data acquisition and full scale processing of the resulting data on a large computer was triggered by the desire to fully exploit the output of a high resolution mass spectrometer used in the determination of structures of complex organic molecules(1)*.

---

[*] It should be noted that computers had been used previously in the computations required to perform routine quantitative analyses on complex mixtures, in which the input data were recorded in conventional form and manually transferred onto punched cards or paper tape.

The accurate measurement of the position of a few hundred lines representing a spectrum recorded on a photographic plate using a Mattauch-Herzog type double focusing mass spectrometer, by far exceeded the patience of a densitometer operator and led to the design of a semi-automated densitometer with punch-card output. Further evolution went through the stages of digital tape recording (2) and finally on-line operation of the densitometer (3, 4). Similar techniques were developed for fast scanning spectrometers of the Nier-Johnson Geometry, which do not possess a focal plane. These instruments thus can not make use of a photographic plate and must record on-line into the computer (5, 6). The primary data obtained in either case is a parameter related to the mass of a particular ion giving rise to the signal, as well as its abundance. The conversion of this parameter to mass and further to the elemental composition (the interpretative important information) of the ion will be discussed later.

A second area of mass spectrometry in which the data rate requires sophisticated data aquisition techniques involves the operation of the spectrometer in conjunction with a gas chromatograph. In order to accumulate all the data necessary to identify all the components of a complex mixture, a complete mass spectrum has to be recorded every few seconds during the entire gas chromatogram, which may last 1/2 - 2 hours.Their accumulation, storage and utilization in a relatively short time is practically possible only with the aid of a fast computer having a reasonable storage capacity and certain sophisticated (at least as far as chemical laboratories go) output devices. Fortunately, the nature of the compounds and the information content of a series of spectra on a mixture of compounds separated by gas chromatography, rather than a single substance, make it in most cases unnecessary to determine the elemental composition of all the ions and one can get by with their nominal mass, i.e. the data provided by a conventional, single focusing mass spectrometer (7, 8).

As alluded to above, the on-line recording of a mass spectrum involves sampling the output of the electron multiplier of the mass spectrometer (or the photomultiplier of the densitometer) during the scan at a rate commensurate with the accuracy required for the mass measurement. In high resolution mass spectrometry one wishes to determine the mass with an accuracy in the order of one part per million, which requires high sampling frequency (10 - 50 KH_z) and high precision with respect to the x-axis parameter (time in the case of on-line scanning of the spectrometer, or distance when using a photographic plate). Since the peak profiles are guassian, their center can be calculated easily by the usual methods, such as interpolation of the data points around the maximum, or the centroid. The degree of sophistication with regard to thresholding and the detection of incompletely resolved multiplets (due to different ions falling at the same nominal mass) differentiates simple approaches useful for routine spectra from more complex ones produced from minute amounts of material of complex chemical structure, in which high accuracy in the measurement of the mass of ions of low abundance becomes of great importance.

An example of the type of data encountered in high resolution mass spectrometry and the relationship of the various parameters is illustrated in Table I.

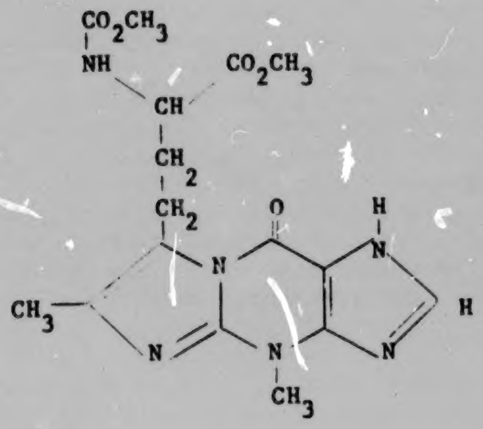
Table I

Position of line center (in mass) measured	Mass calculated (in mass units)	Corresponding elemental composition
173.4863	484.2721	$C_{21}H_{36}N_2O_3$

The relationship between line position and mass is based on the measurement of two lines due to ions of known composition (from an internal standard) and lower and higher mass than the unknown. The elemental composition is computed directly from the mass found to correspond to the line position measurement. It is that combination of elements whose sum of exact atomic weights equals the empirically found mass. The latter has to be determined as accurately as possible, to assure unambigous assignment, because the sum of other combinations may be very close (e.g. the mass of $C_{31}H_{36}N_2O_3$ is 484.27258 while that of $C_{29}H_{34}N_5O_2$ is 484.27123, a difference of only 0.0013 mass units). The same relationship holds for the relationship of the time at which a peak maximum has to be measured in the case of recording on-line with a scanning mass spectrometer. The time differences between the peak center of the reference ions and the unknown ion has to be determined with an accuracy of a few microseconds.

A molecule consisting of so many atoms fragments upon electron impact in the ion source of the mass spectrometer into a large variety of fragment ions varying greatly in abundance. While the more abundant ones are usually also of structural significance, some ions of low abundance may also be of great importance. Thus, the mass of all detectable ions has to be measured accurately and their elemental composition has to be calculated. Considering the large number of combinations of atoms which have to be tested for each accurate mass, a fast computer is the only means by which this can be accomplished routinely. Some of the problems encountered in this work and their solutions have been summarized previously (9).

High resolution mass spectrometry has been of great value in the determination of the structure of certain complex organic molecules available only in extremely small quantities. An example is the recent elucidation of the structure of a nucleoside representing a minor constituent of phenylalanine transfer-ribonucleic acid. The mass of the ion corresponding to the intact molecule revealed its composition to be $C_{16}H_{20}N_6O_5$. A major fragment ion had the composition $C_{10}H_{10}N_5O$ which appeared to be aromatic (few hydrogens) and indicated the presence of a substituent that contains the remaining atoms. The arrangement of the heteroatoms in this substituent could be deduced from certain ions of relatively low abundance corresponding to the loss of some of the functional groups from the molecule. These data, in conjunction with the UV, IR and NMR spectrum of the substance, of which only a fraction of a milligram was available at any one time, as well as synthetic models and biogenetic considerations, led to the proposal of structure I for this biologically significant substance (10, 11).



I

Another important area was alluded to earlier, namely the use of a computer to record, process and utilize the spectra obtained from a mass spectrometer which is monitoring a gas chromatograph. The schematic of such a system is shown in Fig. 1. It is designed for continuously scanning and recording approximately 400 spectra obtained from a GC effluent (7). Data is acquired during the upscan portion of each 4 sec. cycle of the magnetic field. Peak centers and intensity data are determined in real time and the partially reduced data is transferred to disk during the collapse of the magnetic field. Background threshold is continuously updated by the software as the data is acquired. In this way the loss of sensitivity usually associated with arbitrarily set thresholds involved in hardware peak detection is avoided by more fully using the capabilities of the computer.

Presently the system consists of a Perkin-Elmer 990 gas chromatograph, an Hitachi-Perkin-Elmer RMU-6L single focusing mass spectrometer, and an IBM 1800 Data Acquisition and Control System using interface hardware and software developed in this laboratory. The 1800 is configured with 32K of core memory, a 3-drive 1810 disk system, two magnetic tape drives, high speed digital and analog input and output capabilities, a line printer, a console typewriter, and a digital plotter. An analog tape system has been interfaced to the computer and to the various low resolution instruments in the laboratory. This provides the capability for data to be acquired and saved for subsequent digitization and processing by the computer at a later time. Finally, the most recent addition is an oscilloscope-microfilming system.

Improvements in the scanner circuitry to allow a more repetitive computer-controlled scan function and major changes in the software system have permitted faster and more efficient use of the facilities by laboratory chemists. These faster programs have resulted in the increased use of the GC-MS-Computer system for problems in areas such as lunar crust analysis, carbohydrate chemistry, peptide sequencing, analysis of drug abuse problems, geochemical analyses, and various other fields of organic and biochemical interest. With the large amounts of data now generated, it was realized that the limiting factor was the relatively slow speed of the output devices required to present the data to the chemist. Furthermore, it is desirable to make all of the data (not just the spectra) readily available to the chemist without requiring him to return to the computer for each additional piece of information. To meet these goals, a new and faster method of outputting data was sought. The microfilm system described below has efficiently solved these problems.

The microfilming system consists of a Tektronix 611 storage oscilloscope, a Bolex 16mm H16M5 Cine camera with a 10mm Switar lens, and a single-framing control mechanism manufactured by Industrial Camera Company, Epsom, N.H. The plotting and filming are totally controlled by the computer via analog and digital output features. The computer can process and film a GC-MS run of 400 spectra in under 10 min. (including the time-to-mass conversion step). The film is immediately developed using a Kodak Prostar Processor and is thus available to the chemist within a few minutes after filming is completed. The end result is a finished microfilm in a cartridge which can be viewed on the microfilm readers in this laboratory or in any modern library.

The data available consist then of a few hundred consecutively recorded mass spectra. Since a single focusing mass spectrometer is used the spectrum represents nominal mass (rather than elemental compositions) vs. abundance of the ions. These data are then presented in a multitude of interpretatively meaningful ways (12) as outlined in the following examples from work on the organic acids produced upon oxidation of oil shale kerogen (13).

Summing the abundance of all ions in each spectrum and plotting the sum vs. scan index number produces a gas chromatogram (Fig. 2 top; representing 400 mass spectra). The mass spectrum corresponding to any point of interest along the gas chromatogram can be presented (on the scope or plotted) by the computer. For example, scan No. 171, shown in Fig. 3, is interpreted as the methyl ester of the acid with 17 carbon atoms (the formula is not part of the output, but arrived at by the interpreting chemist, or by the computer using the comparison technique outlined below). Alternatively, one can display the relative abundance of a single mass throughout the gas chromatogram, resulting in a "mass chromatogram" as shown in Fig. 2 for mass 88 (middle) and 98 (bottom), characteristic for α-methyl substituted methyl esters and methyl esters of dicarboxylic acids, respectively.

A combination of such display techniques and/or the comparison of the mass spectra obtained during the gas chromatogram with a file of about 10,000 authentic spectra led to the identification of the components in this mixture. They turned out to be unbranched carboxylic acids with 8-29 carbon atoms (C_8-C_{29}), isoprenoid acids with 15-22 carbon atoms (C_{15}^{β}-C_{22}^{δ}, where the superscripted letter denotes the position of the first methyl substituent) and methyl esters of dicarboxylic acids with 7-17 carbon atoms (C_7^{di} - C_{17}^{di}).

The comparison technique mentioned above involves the matching of the significant features of the spectrum of the unknown with each one in the collection and presenting the chemist with the names of the compounds exhibiting the most similar spectra (14). It is up to him to decide whether the most similar one is indeed the same compound or merely the most closely related one. The final result thus rests on the chemist's interpretation of the facts presented by the computer.

Literature References

1. Desiderio, D. and Biemann, K., 12th Annual Symposium on Mass Spectrometry, Montreal, June 1964 (p. 433).

2. Venkataraghavan, R., McLafferty, F.W., and Amy, J.W., Anal. Chem. 39, 178 (1967).

3. Cone, C., Fennessey, P.V., Hites, R.A., Mancuso, N., and Biemann, K., 15th Annual Symposium on Mass Spectrometry, Denver, May 1967 (p. 114).

4. Desiderio, D.M., and Mead, T.E., Anal. Chem. 40, 2090 (1968).

5. McMurray, W.J., Greene, B.N., and Lipsky, S.R., Anal. Chem. 38, 1194 (1966).

6. Burlingame, A.L., Smith, D.H., and Olsen, R.W., Anal. Chem. 40, 13 (1968).

7. Hites, R.A. and Biemann, K., Anal. Chem. 40, 1217 (1968).

8. Sweely, C.C., Ray, B.D., Wood, W.I., Holland, J.F., and Krichevsky, M.I., Anal. Chem. 42, 1505 (1970).

9. Biemann, K. and Fennessey, P.V., Chimia 21, 226 (1967).

10. Nakanishi, K., Furutachi, N., Funamizu, M., Grunberger, D, and Weinstein, I.B., J. Am. Chem. Soc. 92, 7617 (1970).

11. Thiebe, R., Zachau, H.G., Baczynskyj, L., and Biemann, K., Biochim. Biophys. Acta, in press (1971).

12. Hites, R.A. and Biemann, K., Anal. Chem. 42, 855 (1970).

13. Djuricic, M., Murphy, R.C., Vitorovic, D., and Biemann, K., Geochim. Cosmochim. Acta, in press (1971).

14. Hertz, H.S., Hites, R.A. and Biemann, K., Anal. Chem. 43, 681 (1971).

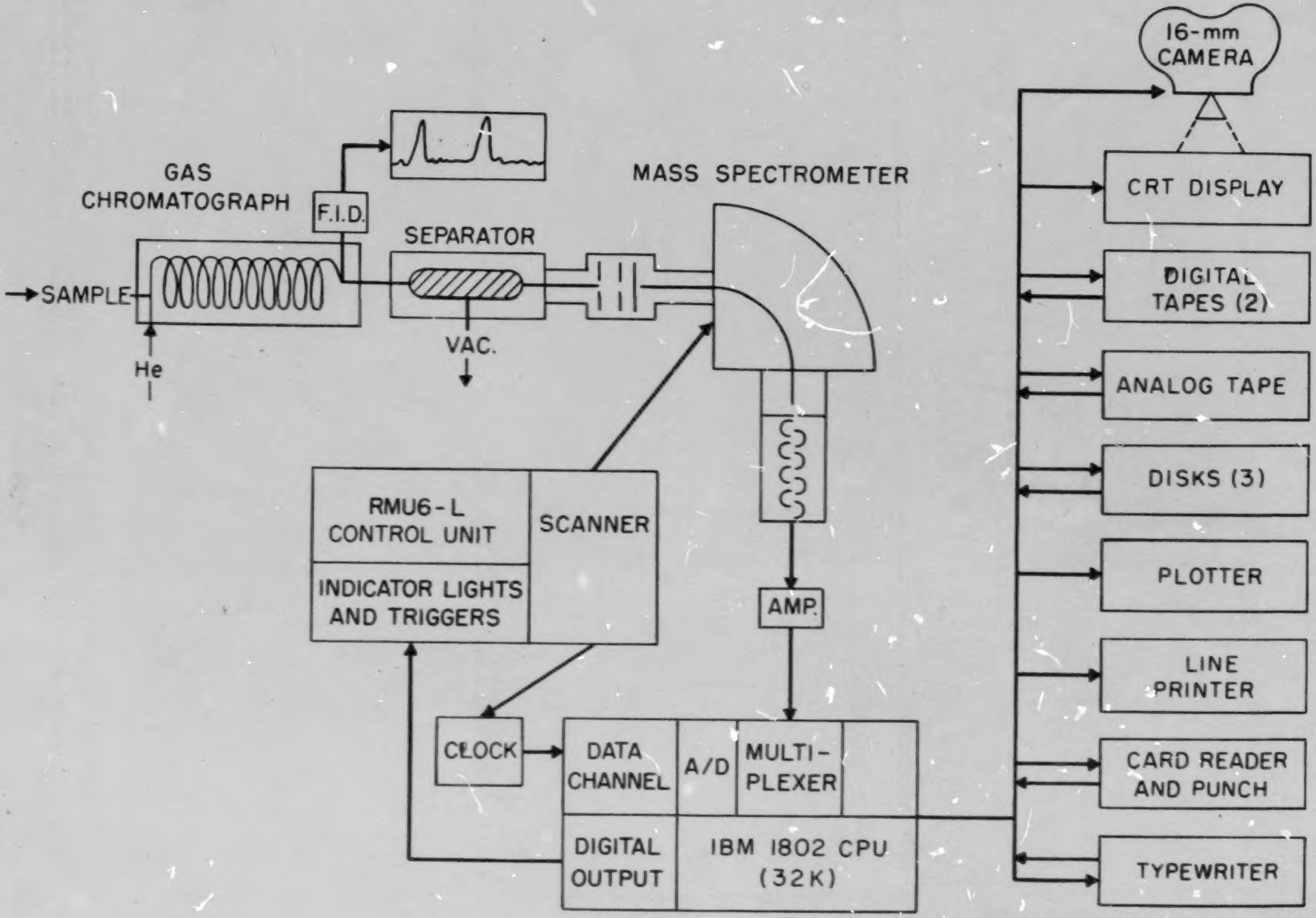
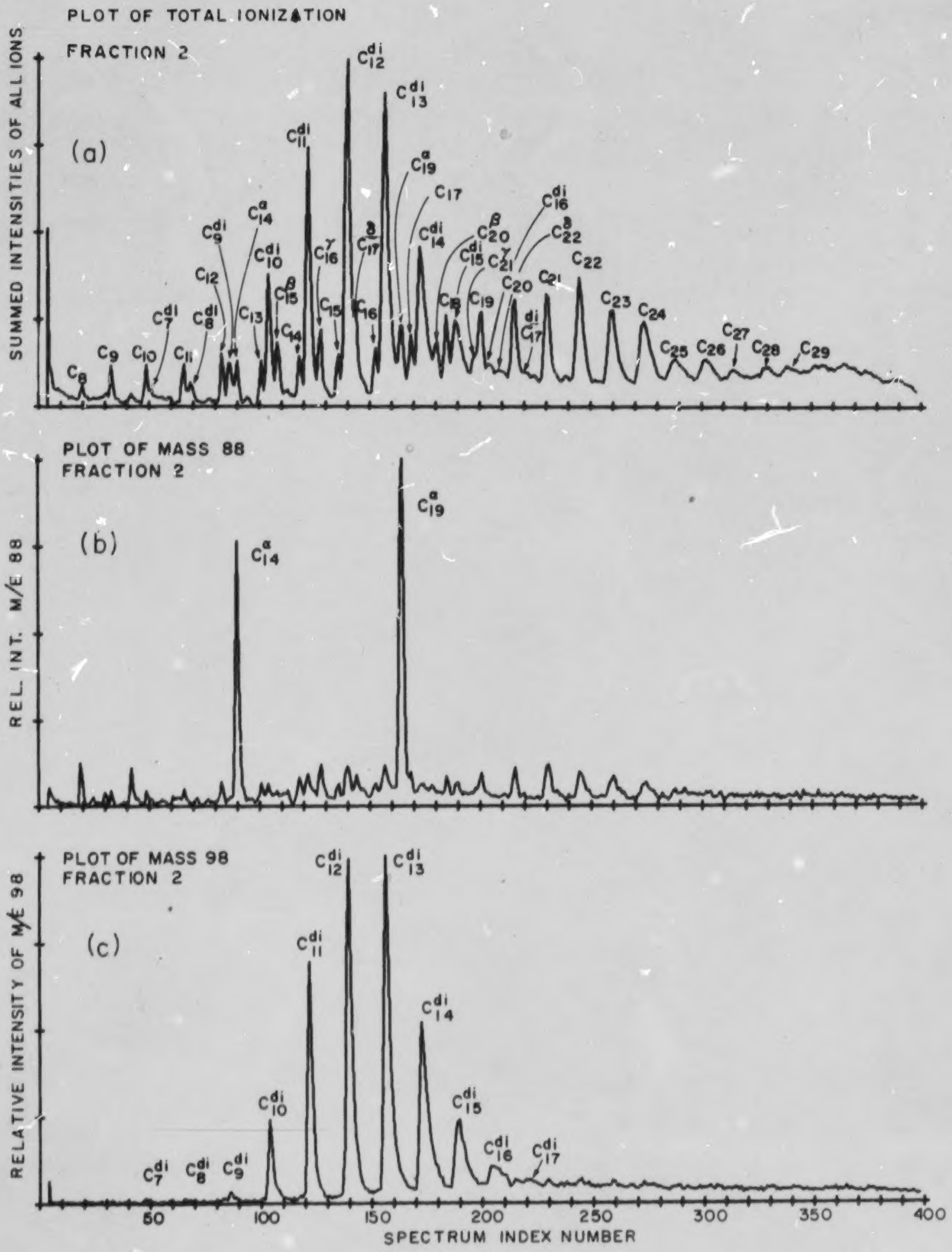

GAS
CHROMATOGRAPH
SAMPLE
He
F.I.D.
SEPARATOR
VAC.
MASS SPECTROMETER
AMP.
RMU6-L
CONTROL UNIT
INDICATOR LIGHTS
AND TRIGGERS
SCANNER
CLOCK
DATA
CHANNEL
A/D
MULTI-
PLEXER
DIGITAL
OUTPUT
IBM 1802 CPU
(32K)
16-mm
CAMERA
CRT DISPLAY
DIGITAL
TAPES (2)
ANALOG TAPE
DISKS (3)
PLOTTER
LINE
PRINTER
CARD READER
AND PUNCH
TYPEWRITER

FIGURE 1



PLOT OF TOTAL IONIZATION
FRACTION 2

PLOT OF MASS 88
FRACTION 2

PLOT OF MASS 98
FRACTION 2

FIGURE 2

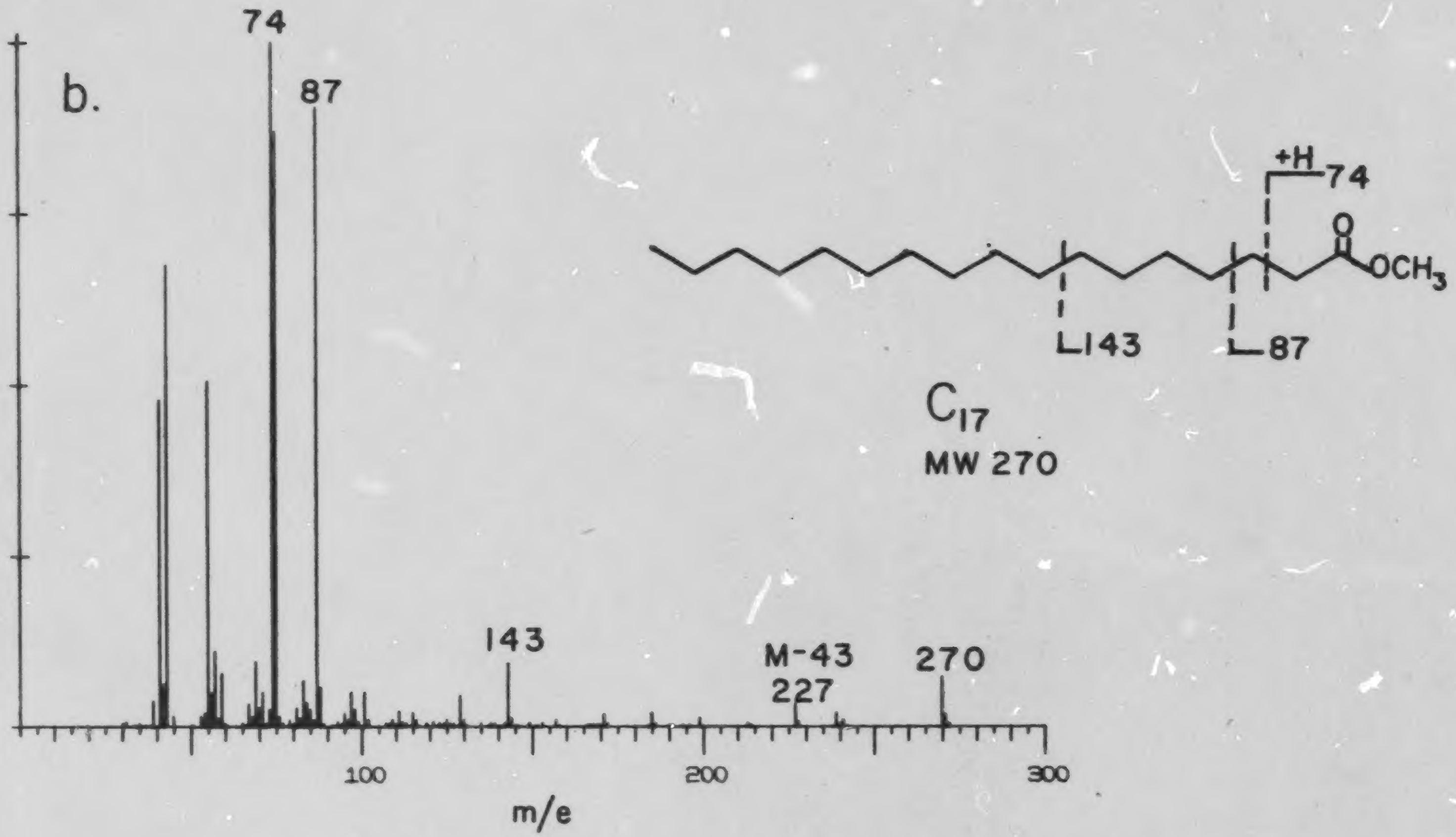

b.
74
87
143
M-43
227
270
100
200
300
m/e
+H
74
O
OCH3
L143
L87
C17
MW 270

FIGURE 3

RESOLUTION OF ELECTRONIC ABSORPTION SPECTRA WITH LOGNORMAL CURVES*

D. B. Siano, J. A. Thomson and D. E. Metzler
Department of Biochemistry and Biophysics
Iowa State University
Ames, Iowa 50010

INTRODUCTION

The resolution of electronic absorption spectra by curve-fitting methods has largely been overlooked while emphasis has been placed on spectra in the ir-Raman region (1). In the latter the bands to be resolved have generally been considered symmetrical: the most popular functions chosen for fitting in a least-squares treatment have been either Gaussian or a combination of Gaussian and Cauchy (Lorentz) curves. Even when asymmetry of the bands was recognized, a "two half-width" Gaussian was applied.

The resolution of spectral bands in the uv region can also provide much chemical information. In addition, resolution provides a means of storing information about absorption spectra in an easily accessible library of uniform format.

THE LOGNORMAL FITTING FUNCTION

Close examination of absorption spectra reveals that in general, individual peaks are not symmetrical. This holds for spectra taken against wavelength or wavenumber, but the asymmetry is more obvious on the wavenumber scale. The bands are generally skewed towards higher energies. The so-called "lognormal" distribution has been found very satisfactory for use in resolution of electronic spectra (2).

This function has the form

$$\varepsilon(\tilde{\nu}) = \frac{\varepsilon_0 b}{\tilde{\nu}-a} \exp(-c^2/2) \exp\left\{-\frac{1}{2c^2}\left[\ln\left(\frac{\tilde{\nu}-a}{b}\right)\right]^2\right\} \; ; \; \tilde{\nu} > a$$

and $\varepsilon(\tilde{\nu}) = 0$; $\tilde{\nu} \leq a$. The molar absorptivity, ε, is a function of the wavenumber, $\tilde{\nu}$, in kiloKaysers (kK), ε_0, the maximum molar absorptivity of the peak and of the four lognormal parameters designated a, b, and c.

---

* This work is supported by Grant AM01549 from the United States Public Health Service.

Three other parameters, termed "practical" in the sense they may be estimated directly from recorded spectra, are defined in Fig. 1. They are the position of the mode, $\tilde{\nu}_0$, the half-width, W, and the skewness, ρ. They are related to a, b and c by the expressions

$$c = (\ell n\rho)/(2\ell n2)^{1/2}$$

$$b = W[\rho/(\rho^2-1)]\exp(c^2)$$

$$a = \tilde{\nu}_0 - W[\rho/(\rho^2-1)]$$

The area, A, under the curve is

$$A = \varepsilon_0\ W[(2\pi)^{1/2}[\rho/(\rho^2-1)]c\ \exp(c^2/2)].$$

For a single spectral band, the lognormal function has been found superior to the "two half-width" Gaussian and provides a suitable analytical expression for describing many spectral bands (2).

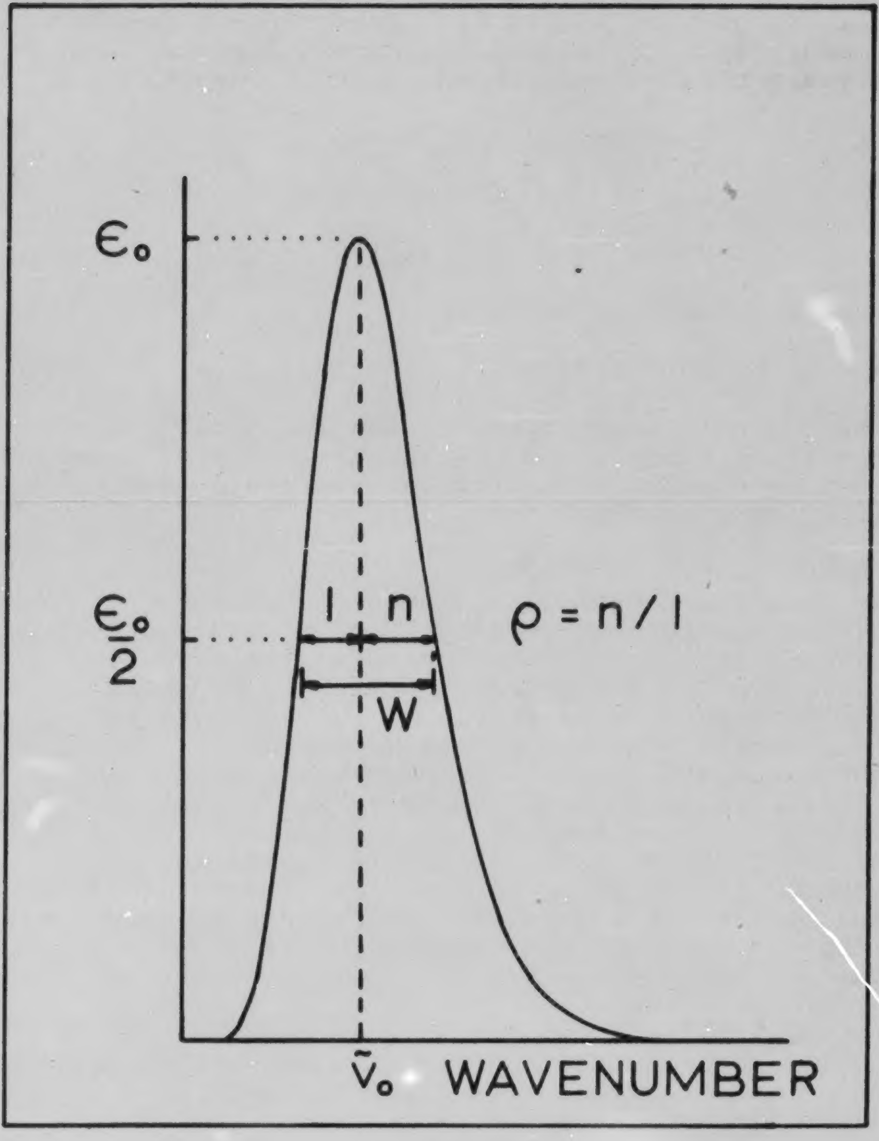
CURVE RESOLUTION

An actual spectrum will generally consist of several bands which may overlap or be buried: the problem of resolution then is one of optimizing many parameters to obtain the "best" fit. Our programs use the least-squares approach and the minimization technique of Fletcher and Powell (3). The coding for this method is available in the IBM Scientific Subroutine Packages as subroutine FMFP. To use this subroutine, the user must supply another subroutine (in our case, DERIV) in which the values of the function and of the error-squared sum, are determined from the current estimates of the parameters and in which the gradients of the parameters are also calculated.

The digital spectrum (usually as molar absorptivities) to be resolved is read in along with initial guesses of the practical parameters for each band which is believed present. The corresponding values of the lognormal parameters are calculated. Control then passes to subroutine FMFP which immediately calls subroutine DERIV. There the sum of the difference at each wavenumber between the experimental and computed spectrum is calculated. The computed spectrum is a summation of the current theoretical values of all the

FIGURE 1

DEFINITION OF PRACTICAL LOGNORMAL PARAMETERS



ϵ_o
$\frac{\epsilon_o}{2}$
l
n
$\rho = n/l$
W
$\tilde{\nu}_o$
WAVENUMBER

individual lognormal bands. The derivatives of the lognormal with respect to the parameters are also calculated from the appropriate analytical expressions and are used to evaluate the gradients of the parameters. The iterations continue in FMFP and DERIV until either convergence is sensed in terms of certain user-supplied values or a given number of iterations is completed. Control returns to the main program and the final values of the parameters and of the gradients together with a comparison between the observed and calculated spectrum are printed.

The output also includes two computer-prepared plots: the first is a trace of the computed spectrum with the actual data points superimposed and the second is a plot of the differences as a percentage of band height.

This program has worked well, but as the complexity of the spectrum increases, the program becomes somewhat inefficient. Experience showed good results were obtained if only two or three peaks were considered at one time. Therefore, the program was modified to consider only part of the spectrum at a time. The new program employs a "sliding-window" to control the number of peaks, parameters and segments of the spectrum considered. The lowest-energy peak which contains at least one parameter to be adjusted is located and this serves as the "base-peak" of the window for this particular sequence. The relationship of this peak to its two higher-energy neighbors is examined and a window-width of one, two or three peaks determined. Then only the parameters to actually be varied are passed to the minimization subroutines. Only the portions of the spectrum within the window are made available to the derivative subroutine.

Upon return to the main program, the adjusted parameters of the base peak are fixed so they remain constant. In further calculations any significant contribution of this peak and similar "fixed" peaks to the spectra within the window are considered in the theoretical summation. The program now moves the window to a new base peak at higher energy. The iterations continue until all peaks have been considered. The greater complexity of this coding is more than offset by the increased efficiency in subroutines FMFP and DERIV.

EXAMPLES

Potassium Iodide. The peak of the charge-transfer-to-solvent spectrum of aqueous iodide ion at 44.2 kK has been fitted with a Gaussian, and the upper portions have been tested and found symmetric (4). However, an excellent fit with slightly skewed lognormal curves is obtained as shown in Fig. 2. The maximum error is 0.7% of the band height and this occurs in the valley at 48 kK. The skewness, ρ, is 1.05 for both bands. Although the skewness is small it is easily measurable (for the 44.2 kK band) and it is clear that the band is not truly Gaussian. For the Gaussian $\rho = 1$, and it can be shown that as $\rho \to 1.0$, the lognormal approaches the Gaussian (2).

5-Deoxypyridoxal. An example of a more complex spectral resolution is shown in Fig. 3. A number of overlapping peaks have been resolved successfully, but this was possible only by assuming values of W and ρ for some of the bands (based on comparisons with related substances) and fixing these. The figure illustrates another feature of the program, the tracing of individual lognormal bands.

3-Hydroxypyridine. Resolution of spectra can be used to advantage to describe effects of changes in solvent or temperature on spectra. Resolved spectra of 3-hydroxypyridine in water and in 30% methanol are shown in Figs. 4 and 5. As indicated in Fig. 4, the neutral form of the molecule exists as a tautomeric mixture, the equilibrium position of which is strongly influenced by solvent composition. This shift in equilibrium is reflected clearly in the spectrum. The first (low energy) band (band I) represents the zwitterion (z) while band II represents the uncharged form (n).

Much experimental work in this laboratory (5) indicates that molar band areas, a°, remain constant with changing solvent composition (and temperature). This being the case, the tautomerization constant, K_z, in a solvent may be evaluated from curve resolution data using the following equation:

$$K_z = \frac{f_z}{f_n} = \frac{a_z/a_z^\circ}{a_n/a_n^\circ} = -\frac{\Delta a_n}{\Delta a_z}\cdot\frac{a_z}{a_n}$$

Here f_z and f_n are the fractions of zwitterion and uncharged forms, respectively; $f_z + f_n = 1$. a_z and a_n are the measured areas of bands I and II, respectively,

FIGURE 2

RESOLUTION OF A SIMPLE SPECTRUM

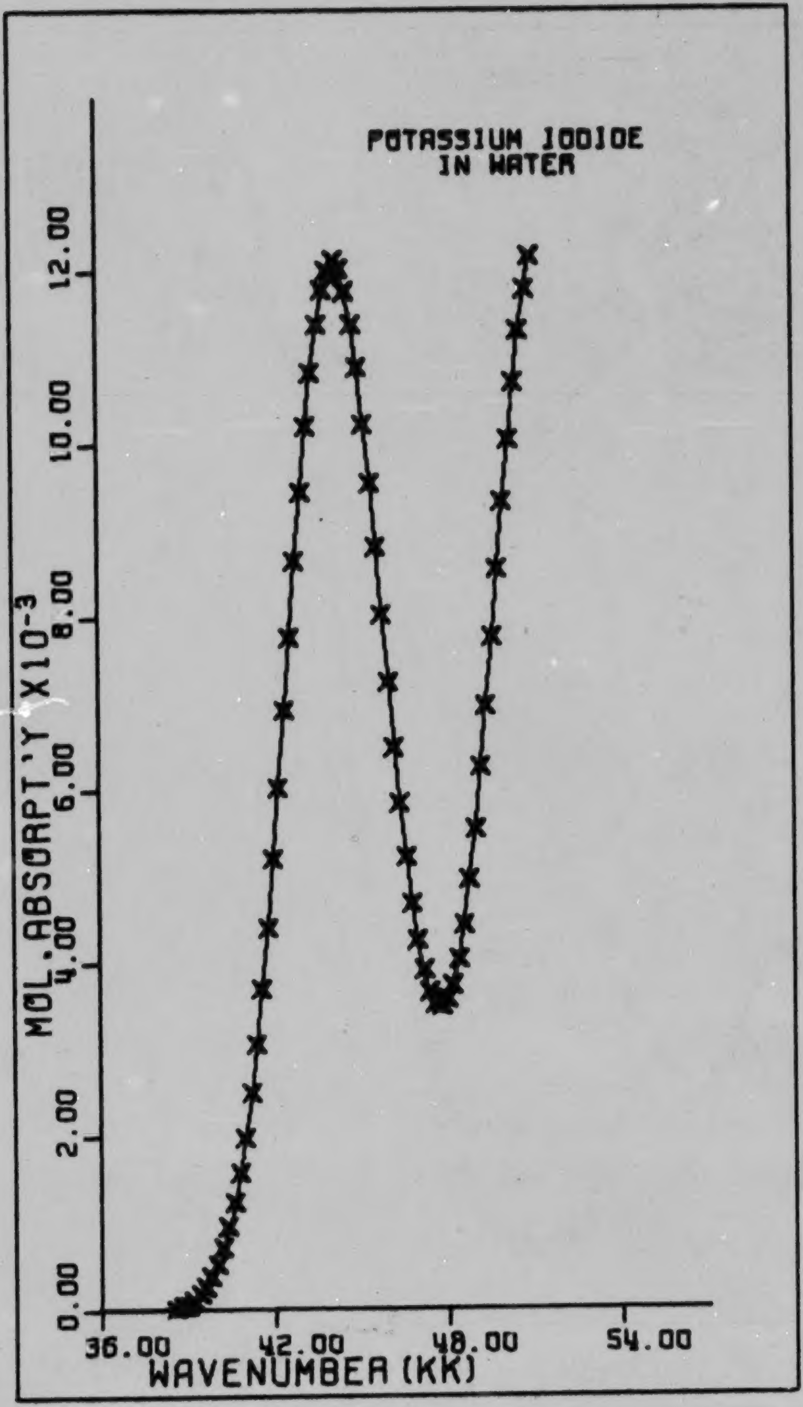


FIGURE 3

RESOLUTION OF A COMPLEX SPECTRUM

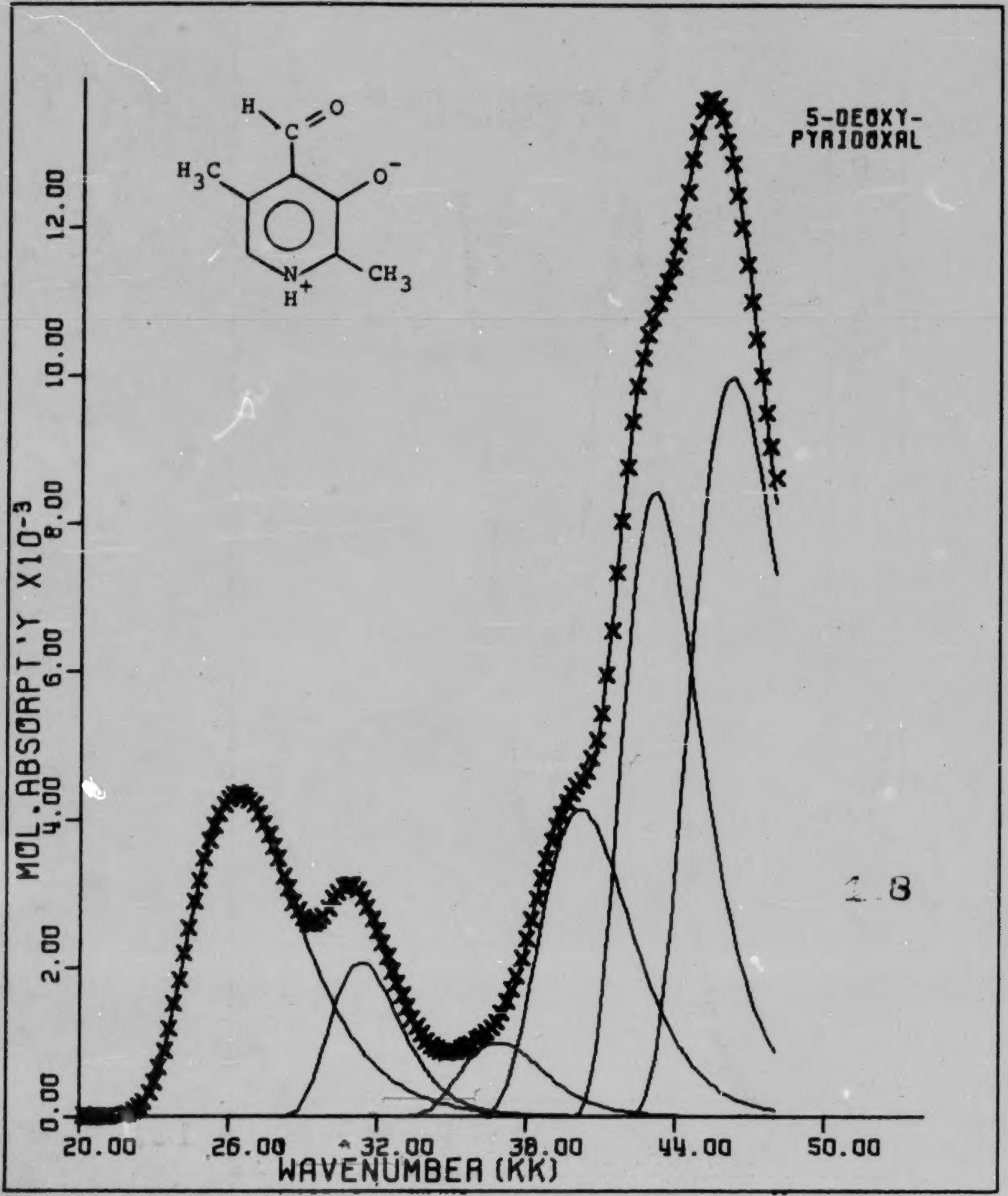


FIGURE 4

RESOLUTION OF THE SPECTRA OF A TAUTOMERIC MIXTURE

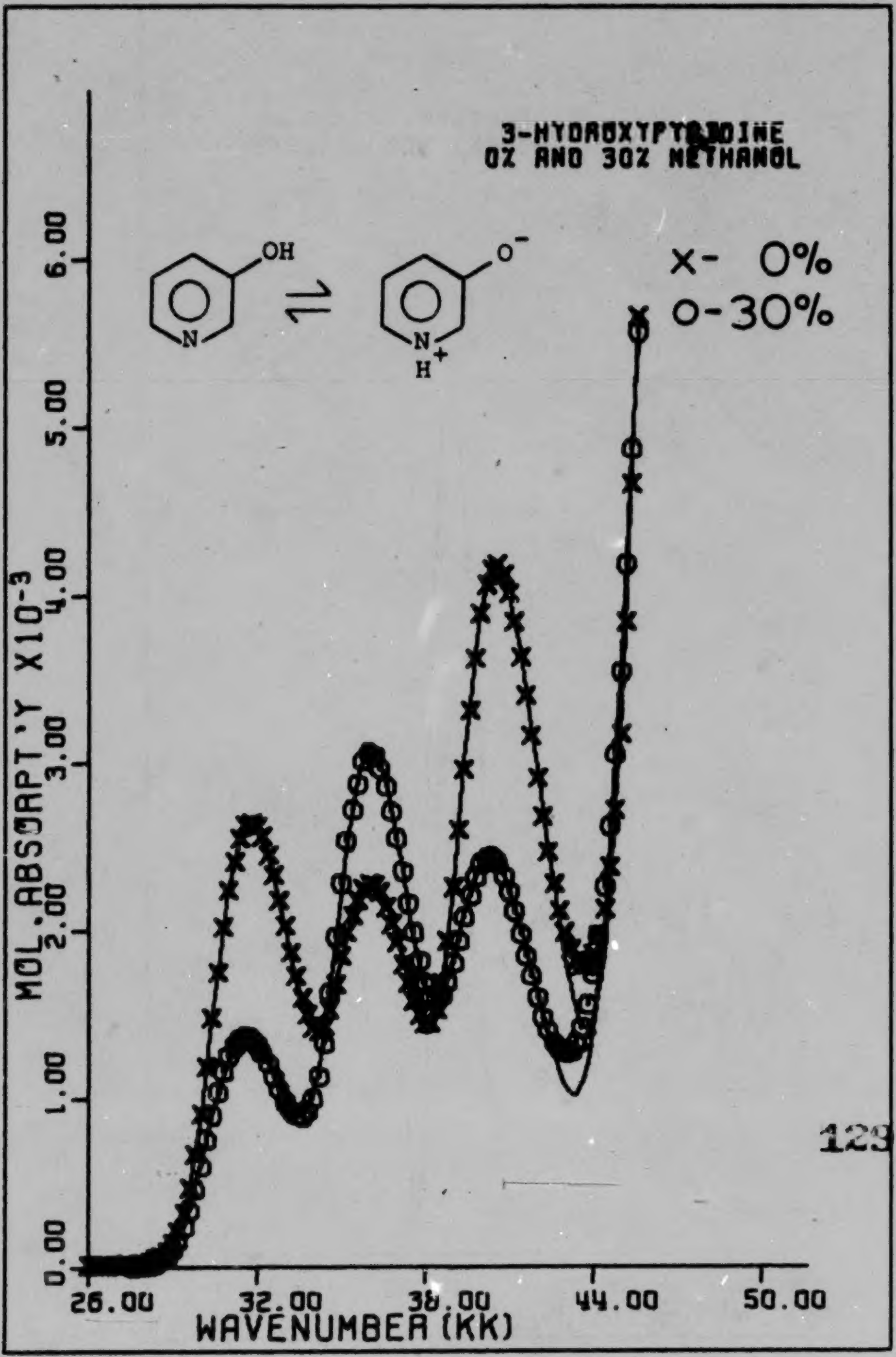
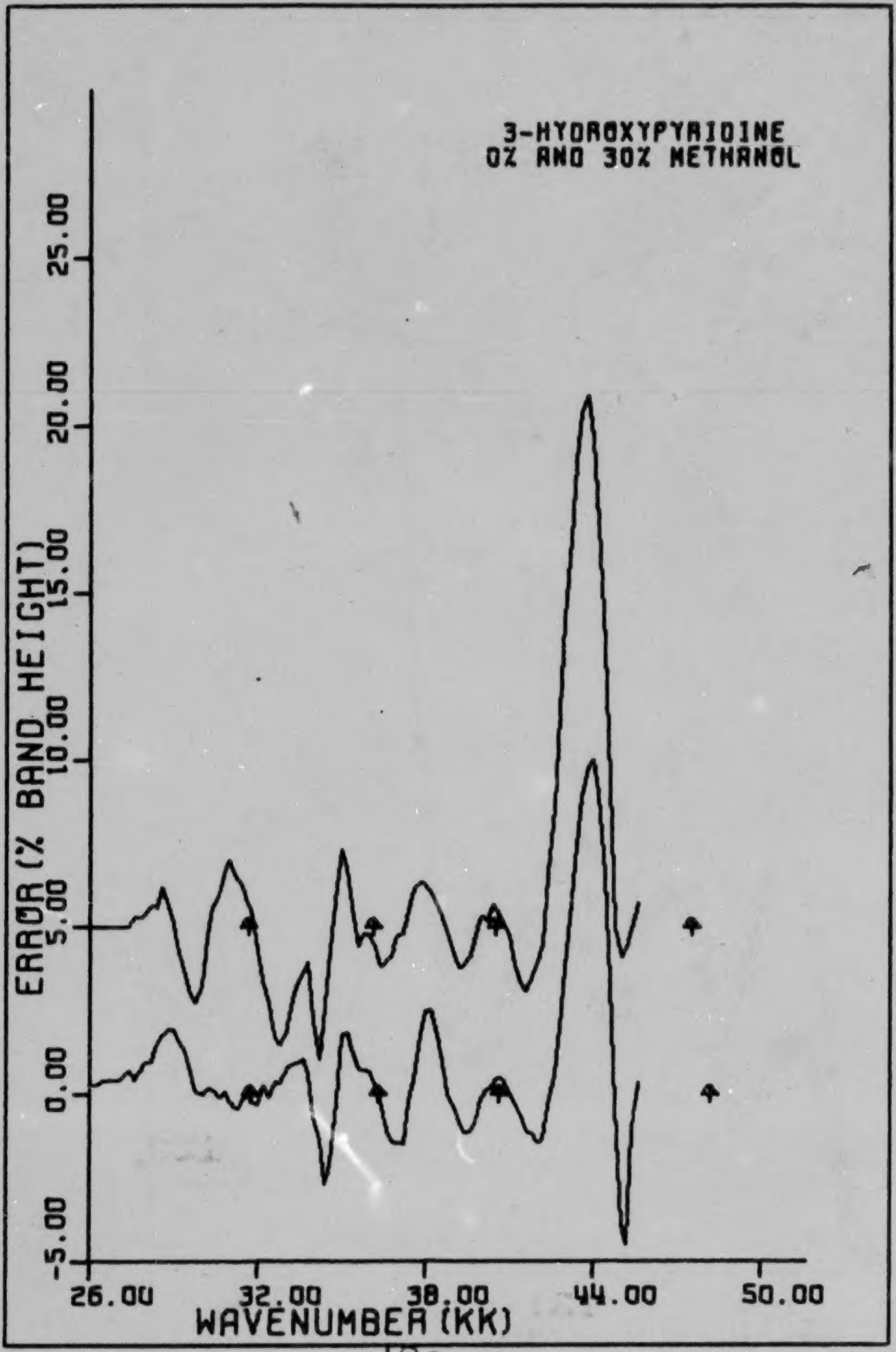


FIGURE 5

DIFFERENCE PLOTS OF RESOLUTIONS OF FIG. 4

Peaks maxima are indicated by arrows.

The bottom curve is for the spectrum in 0% methanol.



and Δa_z and Δa_n are the changes in those areas when the equilibrium position is perturbed, e.g. by a change to 30% methanol. From the data of Fig. 4 we calculate

$K_z = 0.76 \pm 0.1$ in water at 25°C.

The misfit in both spectra at 44 kK is obvious but its cause is less so. We have deliberately weighted the points in the valleys only one-tenth as heavily as the points around the peaks. This insures a good fit around the peaks. Since for some compounds we obtain good fits in the valleys also, it is possible that an additional weak absorption band is present in the spectra of Fig. 4.

The goodness of fit is graphically shown in Fig. 5 which is a plot of the percent error of the two 3-hydroxypyridine spectra. The peak positions have been marked by small arrows. The small shifts in band positions induced by the solvent change are readily apparent and there is a possibility that some of the pattern seen arises from the existence of characteristic vibronic fine structure. This seems especially likely around peak II, for a recurring similar pattern is seen in many spectra.

Benzene. The fine structure of the low energy band of benzene in methanol is clearly displayed by fitting a lognormal curve and examining the difference plot (Fig. 6). This approach can also be used to display fine structure in spectra of proteins or nucleic acid and a way of easily examining effects of perturbations by solvent or temperature changes.

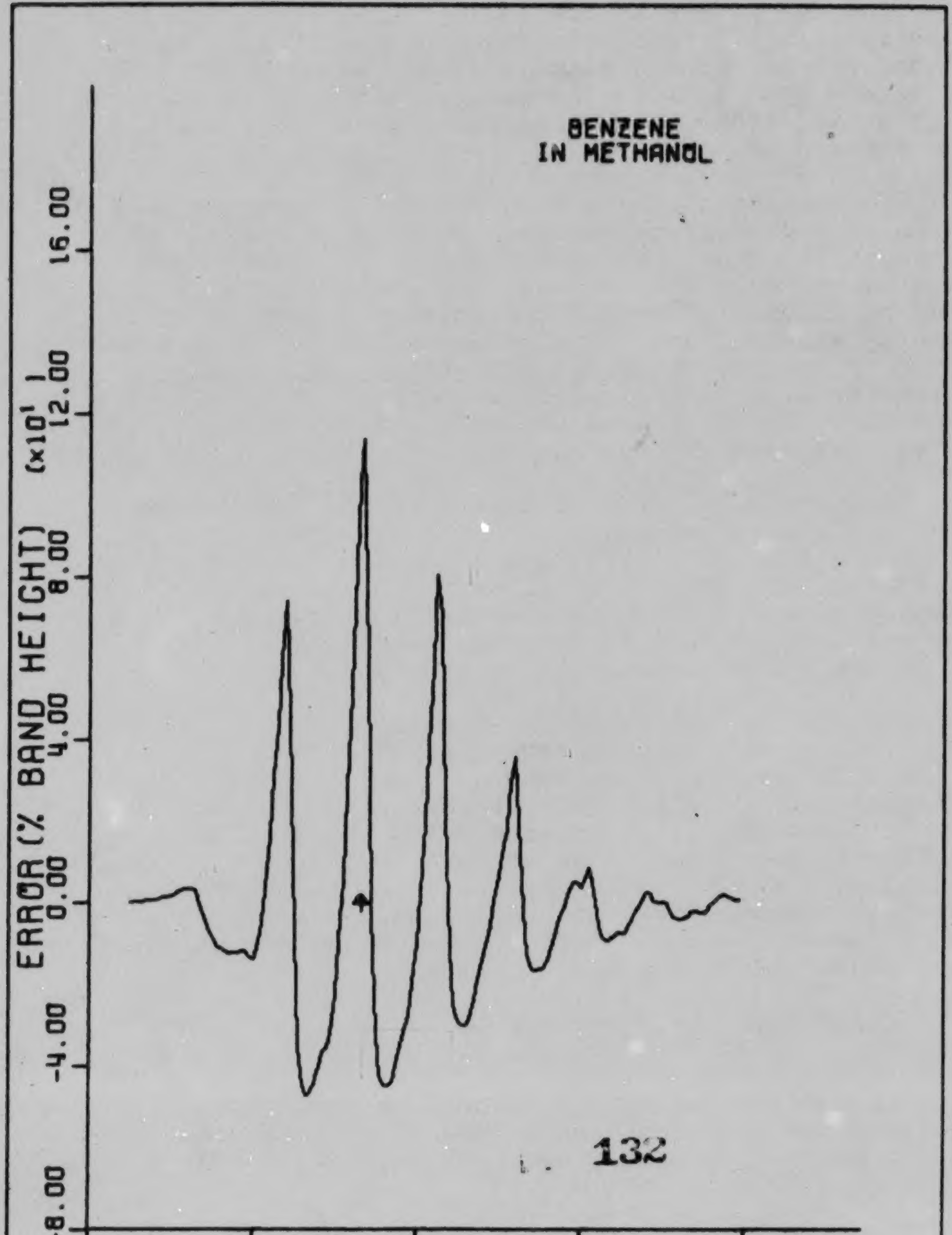
DATA COLLECTION AND PROCESSING

The spectra in this paper were collected in digital form on cards punched at regular wavenumber intervals. A Cary 1501 spectrophotometer was equipped with a Datex digitizing system which is interfaced to an IBM 026 key punch. The absorbance information is sensed by a shaft encoder by specified "wavenumber interval commands" and is punched on the cards in a coded format. A number of programs have been written to process the raw data.

Initially the Datex code is converted to "real" absorbances and baseline, and (if required) turbidity corrections are made. The program allows great variation in spectra-baseline combinations including averaging of repeated scans. The spectra may terminate at different high energy wavenumbers. Small adjustments

FIGURE 6

AN EXAMPLE OF FINE STRUCTURE



for instrumental drift may be made automatically. After temperature corrections are applied, input concentrations may be used to produce decks of apparent molar absorptivities. These decks are then analyzed by other programs to extract equilibrium constants and spectra of individual ionic forms of the compounds (6). The programs feature the use of graphical output and comparison of experimental and calculated data.

A manual describing the complete package of programs is available on request. All programs are written for the IBM 360/65 at Iowa State University in FORTRAN IV using this installation's plotting program SIMPLOTTER on a CALCOMP 470 plotter.

REFERENCES

(1) R. D. B. Fraser and E. Suzuki, "The Use of Least Squares in Data Analysis", to be published in "Physical Principles and Techniques of Protein Chemistry", Part C, S. J. Leach, Ed., Academic Press, New York, N.Y.

(2) D. B. Siano and D. E. Metzler, *J. Chem. Phys.* **51**, 1856 (1969).

(3) R. Fletcher and M. J. D. Powell, *Comp. J.* **6**, 163 (1963).

(4) M. J. Blandemer and M. F. Fox, *Chem. Rev.* **70**, 59 (1970).

(5) D. B. Siano and D. E. Metzler. In preparation.

(6) K. Nagano and D. E. Metzler, *J. Am. Chem. Soc.* **89**, 2891 (1967).

REAL-TIME ANALYSIS OF ULTRAVIOLET ABSORBANCE SPECTRA OF ONE TO FIVE DNA SAMPLES, USING CLASS, A SPECTROSCOPY-ORIENTED LANGUAGE ON AN 8K MINI-COMPUTER

David C. Harris
Institute of Molecular Biology and Department of Chemistry
University of Oregon
Eugene, Oregon, 97403

Overview

The absorbance of ultraviolet light at 260 nm has long been used to study the condition of double stranded DNA and its interactions with other molecules. But existing theory indicates that different nucleotide sequences should have characteristic spectra, and that if one knew the spectrum for each possible interaction, he might derive a wealth of information from analysis of accurate spectral measurements. Rather short nucleotide sequences are being studied by circular dichroism to elucidate the sources* of observable spectra (Cantor *et al.*, 1970; Warshaw and Cantor, 1970). We are working on relatively large DNA molecules, using an empirical but theoretically reasonable method to analyze the spectra. We have developed a system using an 8K Varian 620/I computer and the string-processor language CLASS to control the position of a five-holder sample turret, alter the temperature, and initiate scans automatically on a Cary 14. We read percent transmittance, time, and temperature to high accuracy, punch a paper tape report of the spectrum, analyze the measurements made, and report the most useful results so we can act to change the course of the experiment if desired. The report can be in characters on a teletype, graphically on a plotter, or on a storage oscilloscope. The information reported may easily be varied from experiment to experiment as our chemical questions change. After an experiment we transcribe the paper tape information to magnetic tape and analyze it more thoroughly on an IBM 360/50 H, which has greater storage capacity and sophisticated plotting facilities.

* I wish to acknowledge that Dr. William Melchior, James McGhee, Loren Dotsen, and Chris Meyers did the major share of the hardware development descibed in this paper. I am grateful to Dr. Charles Klopfenstein for advice and instruction in programming and instrumentation techniques, and to Dr. Peter H. von Hippel for generous support and guidance in the physical chemistry of DNA. During the time this research was in progress my stipend was provided by NIH Training Grant PHS2T0100715-12, research support was provided by grant PHS5R01GM15792-05 (to P. von Hippel), and basic computer facilities were provided by PHS-HSAA5S04-FR06027-02.

Theory

Briefly, the spectral analysis method we are using (Felsenfeld and Hirschman, 1965; Hirschman and Felsenfeld, 1966) assumes that hypochromism in DNA arises from interactions between base pairs in a double helix. Thus, an adenine-thymine (AT) pair next to a guanine-cytosine (GC) pair has one spectrum, AT next to AT a different spectrum, and so forth. If the various possible combinations are treated as randomly ordered along the helix, one can deal with all interactions as being due to three kinds of "average" interaction: AT next to AT, GC next to GC, and AT next to GC. Since randomness is assumed, if one knows the fraction AT in a sample he can calculate the fraction of each kind of interaction present. Felsenfeld and Hirschman also introduced a parameter delta to represent small deviations from randomness, so that a positive delta indicates AT pairs being neighbors to AT more often than predicted by complete randomness. They assumed that in a variety of DNA samples the variations in degree of randomness would cancel out. That is, delta is zero overall. They measured the spectra of 11 kinds of DNA, of various fraction AT, and used least squares curve fitting to determine the spectra of the three "average" interactions. This was done for the helical native DNA and for the same samples when completely denatured. The numbers measured were incorporated into a set of convenient algorithms. One measures 4-15 optical densities on a sample under the appropriate conditions, and calculates the total concentration of nucleotides, the fraction AT, and the estimate of randomness, delta. This can be done on a native sample or a fully denatured sample, and the method has frequently been used to establish concentration and nativeness of DNA samples, using manual or desk calculator computations. One can also determine the absorbance changes produced by partial denaturation, and use them to compute the number of base pairs which denatured since an experiment began. Thus, one can follow the melting of AT and GC pairs separately, even though they are present in the same molecule. We are using this separated melting curve technique to study the effects of various aqueous solutions on the two kinds of base pair. A molecule that binds preferentially to regions high in AT pairs may stabilize this region, and we will see the effect as a higher melting temperature for the AT curve.

Hardware

Figure 1 shows the basic components of our system. The central object is our Spectrophotometer Interface (SIF), a modified Cary Instruments option. The SIF is controlled as two separate devices. The relay driver controls a box of power relays.

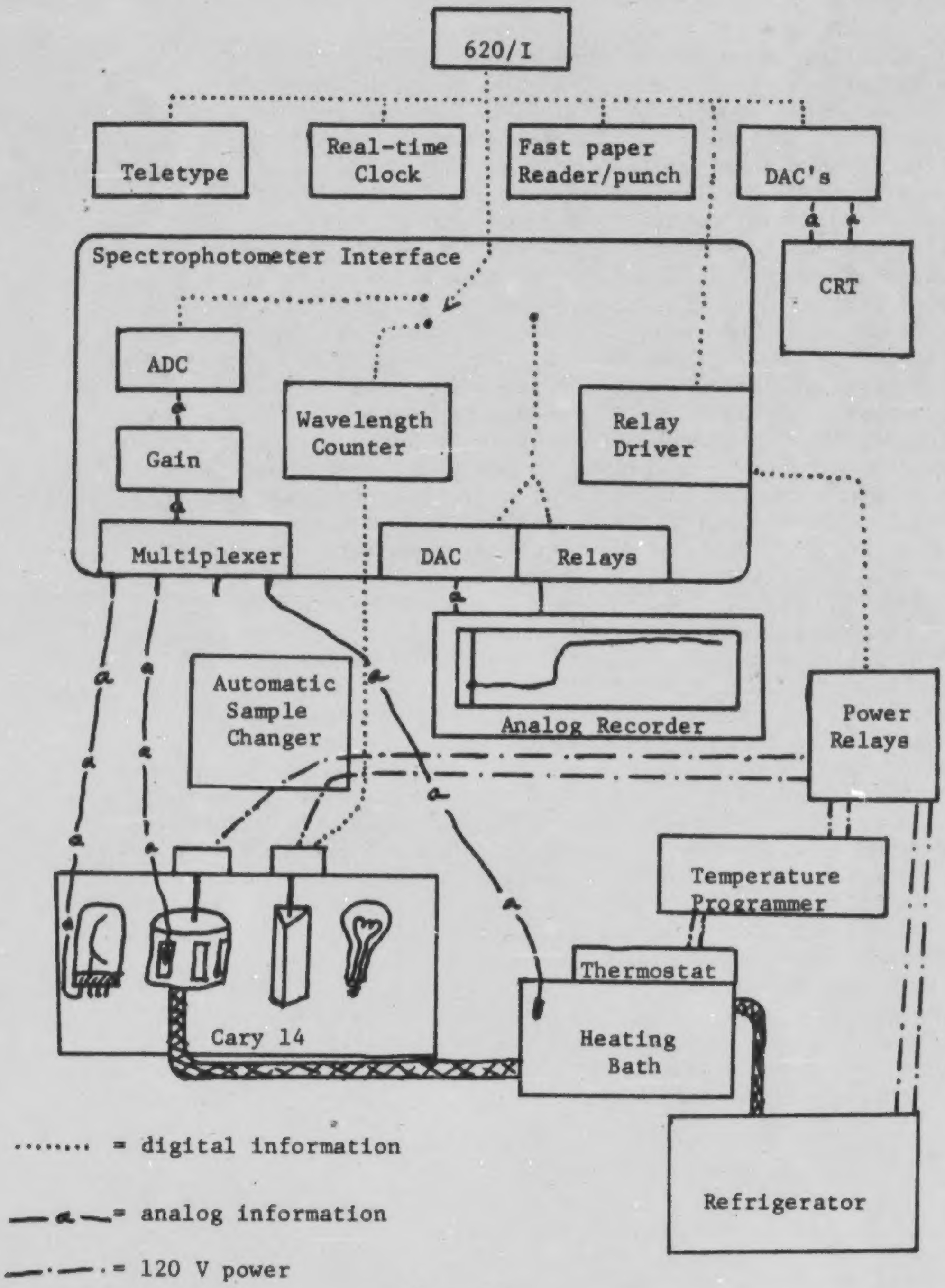

620/I
Teletype
Real-time Clock
Fast paper Reader/punch
DAC's
CRT
Spectrophotometer Interface
ADC
Gain
Multiplexer
Wavelength Counter
Relay Driver
DAC
Relays
Analog Recorder
Automatic Sample Changer
Power Relays
Temperature Programmer
Thermostat
Heating Bath
Refrigerator
Cary 14
a
.... = digital information
—a— = analog information
—·—· = 120 V power

Figure 1

We use some relays to control an Automatic Sample Changer. Our five-sample turret is modified to permit reversing the direction of turn so as not to wind up the thermocouple which goes into one cuvette. Other relays control a modified temperature programmer. This programmer drives the thermostat on a heating bath which continually circulates ethylene glycol to the multiple sample turret. We considered controlling the heating coil in the bath from the computer but decided the bath design probably would not allow us increased stability under direct computer control. The refrigerator was wired to allow on-off and high or low cooling rate selection. Finally, we use the relay box to start the Cary Repetitive Scan option that drives the monochromator over a mechanically set range of wavelengths. The wavelength counter in the SIF keeps track of the monochromator mechanism to 0.01 nm. A portion of the amplified phototube signal goes to one terminal of a multiplexed ADC with selectable gain.

The mechanical drive of the monochromator cannot stop precisely at a particular wavelength. So we begin measuring percent transmittance a little above each desired point and average all readings (at the rate of 30 readings/second) until the monochromator is an equal distance below the intended wavelength. Increasing the width of this "wavelength filter" reduces noise effects by increasing the number of readings averaged. Except on an absorption plateau, an increased filter width also includes more readings unrepresentative of the measurement. We currently use ± 1 nm, which allows repeated scans taken at 2.5 nm/sec scanning rate to have a corrected standard deviation of 0.001 O.D. units for readings below an optical density of 1. We collect such readings every 5 nm over the range 350-220 nm. We can measure temperature to about 0.1 degree C using the thermocouples and calibration tables stored in the computer. We routinely measure time to 1/10 second with the real-time clock. We can punch paper tapes on a teletype punch at 10 characters/sec or on a fast punch at 120/sec. After analyzing a spectrum we report the results on a teletype or storage oscilloscope. We did simple wiring to allow the DAC and relays inside the SIF to control an analog strip chart recorder. Thus we can create simple graphical plots while an experiment occurs.

Software

CLASS (Conversational Language for Spectroscopic Systems) is Cary Instruments' string processor language for their group of spectroscopic instruments. Since this language is rather different from languages commonly used in batch calculations on large computers, we will discuss it here. Figure 2 is an abstract schematic of the language, and figure 3 is a representative

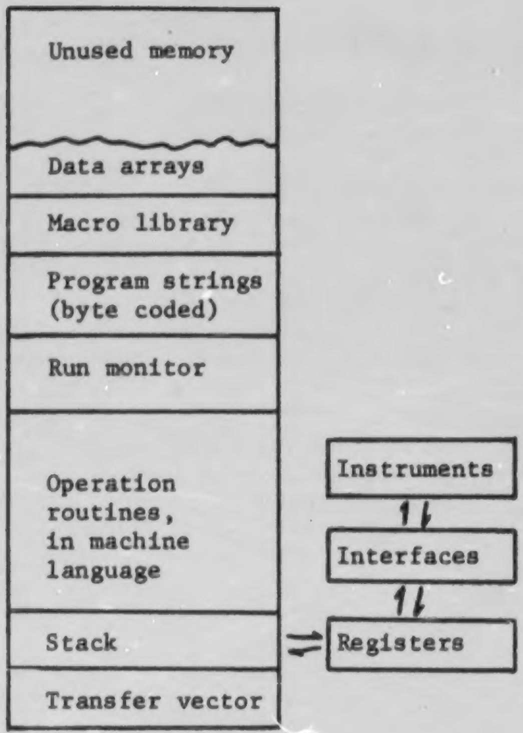

Unused memory
Data arrays
Macro library
Program strings (byte coded)
Run monitor
Operation routines, in machine language
Stack
Transfer vector
Instruments
Interfaces
Registers

Figure 2

```
CALL
 STRG 170
ABSORBANCE
 SPEC 191
 DEST 192
 STRT 35000
 END  22000
CALL
 STRG 164
STRING
.
R95
A36
R32
A192
R22
R51
R15
R33
A192
C35000
C22000
.
M
```

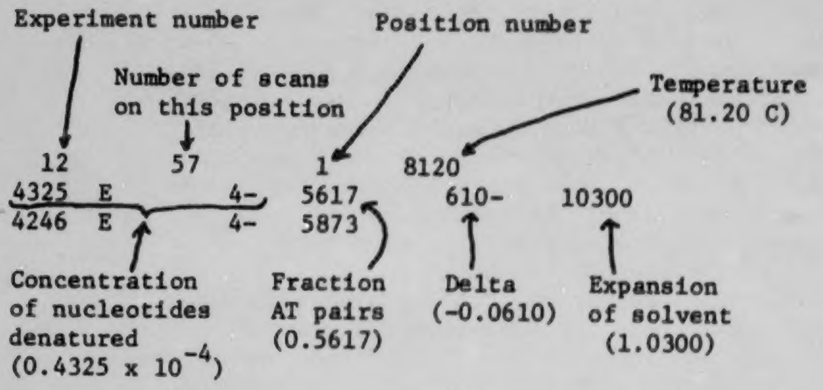
Figure 3

program segment. About 3K of the memory is occupied by the CLASS run monitor and related subroutines. Programs are entered by typing commands which designate particular operations. These commands are stored mainly as a string of 8 bit bytes. During execution the run monitor gets a byte and transfers control temporarily to whichever routine does the desired operation. That much is rather like a second level of machine language. An important feature is that the contents of memory are "floated". That is, information is moved up and down in memory, and some addresses are altered to allow insertion and deletion of routines, programs, and data. A "transfer vector" of addresses is maintained, so one can get to any desired routine by doing an indirect jump through the transfer vector. This strategy assures complete utilization of memory space. By limiting the number of operation routines to fewer than 150, and using bytes for coding, CLASS makes program code very compact. CLASS saves more space by frequently using the same operation routines to manipulate spectral measurements and program instructions. After all, both are linearly arranged and sequentially accessed. Because there is no compile step, CLASS functions conversationally and program modification is fairly easy.

CLASS has a set of "macro" commands for the most common operations, like taking a spectrum, subtracting a baseline, printing a label, etc. Naming a macro causes the appropriate bytes to be moved from a macro library into the program string. Macros being moved can request the programmer to enter the particular values needed for parts of the program that are variable, such as the wavelengths to be scanned, where to store the readings, etc. A programmer can write useful macros for his own common operations and put them in the macro library using a macro compiler feature. Recent versions of CLASS have added multiple program strings, inter- and intra-string conditional jumps, byte manipulations and floating point arithmetic. The jumps allow us to write program string subroutines. The various levels of subroutine allow us to write highly modular code for easy modification. Since modularity increases the possibility that something will be missing when execution is attempted, CLASS provides a number of error messages which often indicate what was wrong or missing. CLASS' interactions with the various instruments are isolated in a few operation routines that read or drive the peripheral hardware. Most I/O occurs through the registers and the push-down stack. The stack is also where all arithmetic is done, using Polish notation sequences of operations on the top two locations.

CLASS has a feature to make handling spectra convenient: the sentence. A sentence is all program code between two periods (.). If a sentence contains any operation that should be applied to every datum of a spectrum, such as putting the sum of two spectra into a third data array, the run monitor causes the sentence to be reexecuted once for every point in the spectrum. The sentence is somewhat like the FORTRAN DO loop, but the programmer need not know the number of points in the spectra being handled.

The main price of these advantages seems to be speed, particularly in arithmetic calculations. For most chemical instruments this is no problem. Since we do extensive arithmetic in the spectral analysis, we occasionally see delays due to calculations. Our monochromator mechanism takes a minute to rewind, so we have sufficient time for all calculations, I/O, and control operations. After we collect and store baselines, a typical program might 1) position the desired cuvette in the light path, 2) start the monochromator and measure time, percent transmittance, and temperature, 3) punch a paper tape of the basic measurements, 4) increment counters of spectra taken, 5) convert percent transmittance to absorbance, 6) adjust for the effect of solution expansion, 7) subtract a baseline, 8) analyze the spectrum and store the results, 9) compute and report the desired function of the basic results, 10) control the bath temperature and 11) increment the cuvette position counter before returning to step 1). The system can run unattended, but we attempt to watch for unexpected results that require further observation or a program change. Figure 4 represents the kind of printed report we can produce.



```
12        57        1    8120
4325  E       4-  5617      610-    10300
4246  E       4-  5873
```

Experiment number
Number of scans on this position
Position number
Temperature (81.20 C)
Concentration of nucleotides denatured (0.4325 x 10^{-4})
Fraction AT pairs (0.5617)
Delta (-0.0610)
Expansion of solvent (1.0300)

The third line of the report is the result of the two term analysis, a computationally simpler approximation to the full spectral analysis.

Figure 4

For the Future

Our current system has two distinct weaknesses: all the paper tape is bulky and inconvenient, and the large computer processing, a straightforward batch calculation, does not occur until after the experiment is finished. Dr. Klopfenstein of the Oregon Chemistry Department has discussed the possibility of a "data sink" system. One computer with a magnetic disk unit would be centrally located and connected by permanent wires to terminals in most rooms. We could connect to the nearest terminal and send spectral readings to the "sink" much faster than paper tape can be punched. Then we would dump our readings from the disk onto a magnetic tape compatible with the IBM 360. This may never occur because of an even more promising possibility: mini-computer communication with a recently acquired time sharing PDP-10 computer. We will not need paper or magnetic tape when we can send information to this large computer. Such a system would make elaborate analysis feasible, with the results returning in time for experimental control.

References

Cantor,C.R., Warshaw,M.M., and Shapiro,H., (1970),Biopolymers 9, 1059-1077.

Felsenfeld,G., and Hirschman,S.Z., (1965), J.Mol.Biol. 13, 407-427.

Hirschman,S.Z., and Felsenfeld,G., (1966), J.Mol.Biol. 16, 347-358.

Warshaw,M.M., and Cantor,C.R., (1970), Biopolymers 9, 1079-1103.

Applications of Computers to Structure Determination and Kinetics

Introduction

Walter C. Hamilton

Chemistry Department, Brookhaven National Laboratory,
Upton, New York 11973

Westheimer has defined the three main areas of concern in chemistry as synthesis, structure, and dynamics. During this conference, we have heard of the many contributions that computers have made in each of these areas. The papers in this session represent a few of the many existing applications to the latter two areas--structure and dnnamics. In fact, most of the papers in the other sessions are directed toward these areas in one way or the other. Today's papers provide examples of each of the several ways in which computers have had an impact on chemistry and on the ways of doing chemistry: large-scale digital computations, on-line experimental control, and chemist-computer interactive coupling.

The use of computers in structure determination work has grown in many areas, but the most dramatic effect has perhaps been in crystallography. Because of his great, easily-formulated needs, the crystallographer was the first major user of the modern digital computer. It is no accident that one of the authors of FØRTRAN was a crystallographer and that former crystallographers often hold important positions in computer systems development. Our first invited paper in this session describes something of the impact that the computers have had in this field and of how the chemical crystallographer uses the computer. The renaissance in inorganic chemistry and the rapid propagation of the faith in molecular biology have both been largely due to the kind of crystallography that computers have made possible.

Following Professor Ibers' contribution, we have two short papers on quite different applications of computers to structural investigation. The first contributed paper describes a correlation between optical spectra and stereochemistry which has been explored by model computer calculations. The speed with which the computer can explore and test hypotheses in a multiparameter space is here the key to its application.

The second contributed paper shows how computers have been used for data acquisition and experimental control in microwave spectroscopy, one of the many fields where the computer has become an important piece of chemical equipment.

We continue to explore the use of computers in experimental control and man-machine interaction as we turn to the invited paper in chemical kinetics by Professor Wilson. Both kinetic experiment and theory have benefitted from the use of the computer, and in the two contributed papers we hear of the use of computers in the analysis of experimental rate data.

It is also worth noting that Professor Wilson and others have contributed greatly to the success of this conference and to the educational use of computers by their elegant computer-produced films of reacting atoms--real kinetic art.

The important point about the use of computers in both of these areas is that more chemists are able to do more interesting chemistry because the computer can do many things much faster than can a chemist--hopefully leaving the chemist more time to think about the meaning of his results.

THE USE OF COMPUTERS IN CRYSTALLOGRAPHY

James A. Ibers, Department of Chemistry
Northwestern University, Evanston, Ill. 60201

Introduction

For the purposes of this paper I mean by the term "crystallography" the determination of the structures of crystalline solids based on data obtained by diffraction methods from single crystals.

The general technique of single-crystal structure determinations has undergone a marked increase in importance in the last decade. A glance at any number of chemical journals, for example Inorganic Chemistry, will reveal that this technique is at the heart of perhaps 15-30% of the papers in a given issue. Moreover, if one follows the trends that have occurred over the last ten years it is clear that more people are now doing larger structures more accurately. There are two main reasons for these changes: (1) the availability of fast, number-crunching computers and (2) better instrumentation. In the instrumentation area, the crystallographer is making extensive use of small, control computers. These are thus the two aspects of computers in crystallography that I will discuss.

Number crunching

Turning first to straight number crunching, the crystallographer envisions calculations that will tax most any available computer. His calculations are generally unsophisticated, well-behaved, but exceedingly arduous - just right for the big computer. Without describing in any detail how the crystallographer determines crystal structures, one can state simply that the process may be divided into several parts:

1) Data reduction

2) The trial structure

3) Structure refinement

In a typical determination the crystallographer will collect between 1,000 and 5,000 data. These will be reduced in a straightforward manner by corrections for background, instrumental factors, etc. This type of calculation is routine and rapid. It is at this stage that the data will be entered into the big computer through cards, paper tape, or magnetic tape, and in most installations from then on the data will be kept either on magnetic tape or on disc files. It is neither practicable nor economical to use card in/card out procedures when there are 5,000 data points. Moreover, because of the large number of data, the calculations in crystallography are not well suited to remote terminals unless high capacity linkages are available.

The finding of the trial structure may require considerable non-routine computing in which human decision making is necessary. But this area has developed significantly in the last decade, and there is the definite possibility that many structure determinations will be handled in an increasingly automatic manner in the next few years. Among the calculations involved in the finding of a trial structure are three-dimensional Fourier summations, where the resultant function is evaluated at perhaps 50,000 points based on a summation over the 1,000 to 5,000 data points. It is interesting to note that this type of calculation required about 40 man-hours in 1950. Today it can be done in a minute or two on a CDC-6000 class computer. In addition the results can readily be plotted on a graphics device.

Refinement of the structure uses up most of the computing time necessary for the determination of crystal structures. Generally, except for the biggest structures, refinement is effected through a nonlinear least-squares procedure in which one attempts to adjust m parameters that define the structure, based on the agreement between calculated data points and the n data points obtained experimentally. Such a procedure works if one is sufficiently close to the correct solution to solve successfully for the corrections in the initial estimates. The calculation requires the inversion of an $m \times m$ matrix. By the nature of the problem this matrix is real and symmetric and hence the elements can be stored in $m(m+1)/2$ cells in the computer. It is instructive to see how this function varies with m:

m	$m(m+1)/2$	Type of Problem
50	1,275	small organic molecule
200	20,100	typical coordination compound
500	125,250	120 atom problem
5000	12,502,500	1200 atom problem (enzyme?)

Thus one can appreciate that since most available computers have only 32K or 64Kwords of high-speed core a severe restriction is currently placed on the refinement of large structures. Thus far, few structures with m past 300 have been properly refined. And one simply does not refine large biological molecules in this manner. Yet the desire to do so remains, and crystallographers look forward to computers with virtually unlimited high-speed cores.

Can one invert such large matrices to obtain meaningful answers? Yes, because the matrices are well conditioned. It has even proved possible to invert these large matrices using the very short word of the IBM 360 series computers.

Where does most of the time go in these refinements? It may be of some interest to note that the rate-determining step is not in the matrix inversion, but rather in setting up the matrix. The operation involved is of the type

$$A = A + B*C$$

and this operation must be performed nm^2 times. On the fastest available computers such an operation might require about 1 μsec. If one has n = 5,000 and m = 300 the time for setting up the matrix is 500 sec., a long time on the very fastest computers. If the time for the operation is 10 μsec., the calculation will take at least 5,000 sec. of central processor time. In a typical calculation on the CDC-6600, a refinement involving n = 3000 and m = 200 will require perhaps 500 sec.

It is clear that the crystallographer not only needs computers with far more high-speed memory but also ones that are much faster. The general rule has been that the faster the computer, the more calculations one can perform per dollar. Although it is difficult to give an average figure, it is probable that most of the well-refined structures reported in Inorganic Chemistry involve about 5K$ of computing per structure. Thus if the single-crystal structure determination is going to become generally feasible as an analytical technique it will have to await the next generation of faster, and cheaper per calculation, computers.

Control aspects of small computers

In the acquisition of data the crystallographer basically performs a simple experiment. He orients his single crystal with respect to the incoming and

and outgoing radiation beams. To do this he drives three or four motors to precomputed positions. He then performs some type of measurement, perhaps over an angular range or for a fixed time. These operations are ideally suited to automation, and to control by a small computer. Within the last four years a number of commercially available instruments have come on the market (generally at 80K$ to 120K$) in which the basic control device is a small computer. The pioneering work in this area was performed by W. R. Busing at Oak Ridge and his experience has been that a computer with a 4Kword core can perform the task.

In such operations the small computer is not being used much of the time, as the real time of the experiment is slow. For this reason there has been considerable effort lately to have the small computer perform reduction steps on the previous data point while handling control of the collection of the present data point. To do this in general one requires more tha 4Kwords of core, but discs are generally inexpensive additions to small computers. However, by the time one buys a small computer, a disc, and perhaps an industry-compatible magnetic tape output device, one has invested about 30K$ in the "small" computer. Perhaps this is why the manufacturers of small computers do so well.

Where do we go from here

The advancement that has been obviously needed for several years now is a direct link between the experiment and the large computer. The small control computer is simply inadequate to perform all the desired tasks, and if it is adorned with myriad accessories the calculations, even if feasible, still are prohibitively expensive. If a link between the experiment and the big computer existed then one could not only check a current measurement against one made several thousand operations before, but one could anticipate a routine structure solution during data acquisition. Once the overall structure is known it is possible to decide which data points are most critical for the accurate determination of a structural feature of chemical interest; in this interactive mode the big computer could direct the course of data collection.

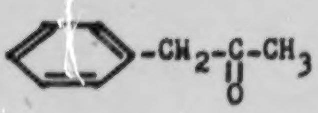
Does this sound impossible? No, it does not to those of us who have grown up with computers. To me it sounds far more likely than did the notion 20 years ago that the calculations I was then performing in 40 hard hours of my time on an IBM 604 electronic card punch would some day (the present) be done in less than a minute, and at a far lower price.

3-7

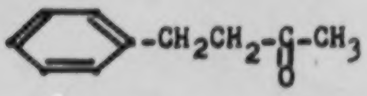
A MODEL STUDY OF KETONES WITH ROTATING GROUPS

S. MacKenzie, S. F. Marsocci, and Mary E. Scanlan
Pastore Chemical Laboratory and The Computer
Laboratory, University of Rhode Island

In phenylacetone (I), there is electronic interaction between the carbonyl group and the benzene ring under the influence of light. The ultraviolet spectrum therefore shows a great increase of absorption (fig. 1). In benzylacetone (II), the probability of effective cooperation is much reduced so that each group shows its normal uv absorption.



I



II

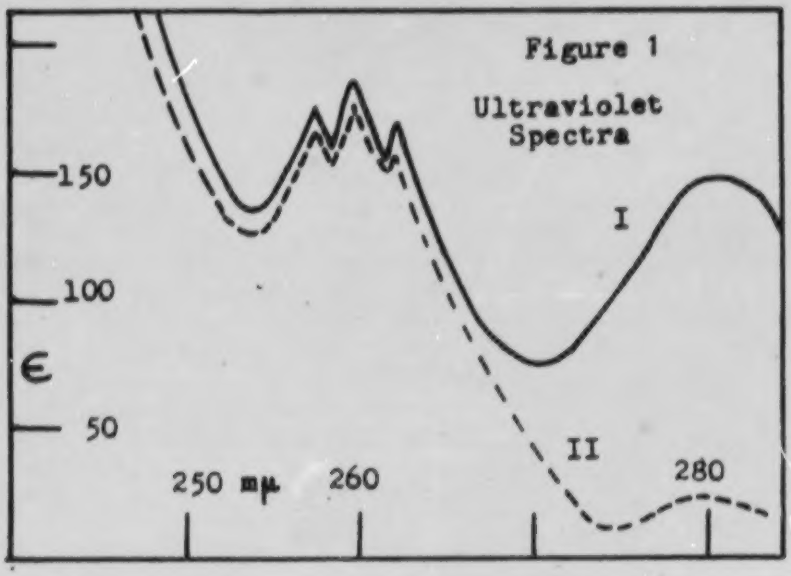
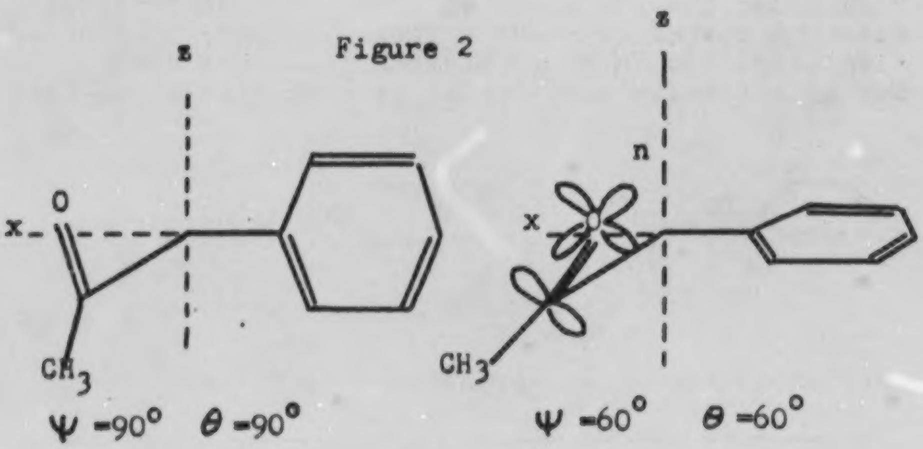


Figure 1
Ultraviolet
Spectra

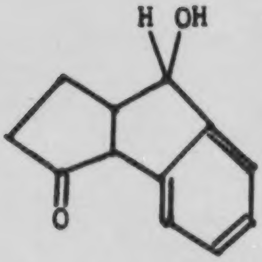
Several reasons for this electronic interaction have been proposed in the literature[1]. Cited most widely are the ideas of Cookson[2] and Labhart and Wagniere[3]. In the former, it was thought necessary for the plane of the carbonyl group to face the plane of the benzene ring as in figure 2a. In the latter, participation of the n-electrons on oxygen was suggested. This idea (in our hands) led to strongest interaction when the oxygen atom was nearer the benzene ring as in 2b.

Figure 2



2a 2b

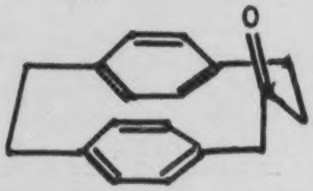
Although many α-phenylketones have been synthesized, the role of geometry in controlling the ultraviolet spectra has not been firmly elucidated. After all, the angular difference between 2a and 2b is not terribly great and it is usually true that precise bond angles have not been measured. Models, quite accurate for bond distances, serve less well for precise estimation of bond angles. Moreover, certain molecules are intrinsically difficult to represent with models and elements of uncertainty arise even for cases of "fixed geometry". This is certainly true for compound III (the geometry of cyclopentane rings is difficult to represent with models), for IV (the large ring is not perfectly rigid), and perhaps even for V, although errors here are much less serious.



III

60°, 60°

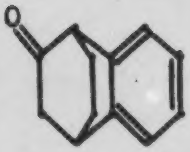
295 mμ (224)[4]



IV

90°, 90°

285 mμ (709)[5]



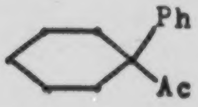
V

120°, 60°

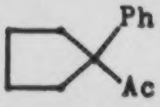
295 mμ (278)[6]

A further source of difficulty is found in the fact that the energy levels of benzene rings are somewhat perturbed as a function of the degree of substitution.

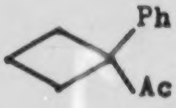
It was therefore felt desirable to study the geometry of α-phenylketones wherein only monosubstituted benzene rings were present. These can be prepared in large number and should not involve any shift in the normal energy levels for the aromatic system. The sole difference in this family of ketones lies in the substituents present on the α-carbon which influence the positions taken by the ketone and aromatic groups.



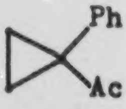
298 mμ (230)



296 mμ (217)



295 m (252)



285 mμ(50)

In early computer programs which represented this situation, each atom had three coordinates (x,y,z) and each of these was expressed, where necessary, as a function of the angles of rotation. A "rotation" was merely a recalculation at the various angular positions. The phenyl ring was "turned" in 15° increments throughout 180° while the ketone group was turned throughout 360° in the same steps. At each position, the energies of interatomic repulsion were calculated. For example, the repulsion energy, R(H-H), between two hydrogen atoms is a function[7] of of the distance, D, between them.

$$R(H\text{-}H) = 10^{4} e^{-4.6D} - 49.2/D^{6}$$

Such equations are invalid at very small values of D and, when these arose, the program caused the setting of population value to zero and the bypassing of further calculations at that angular setting. All the repulsion energies were summed, entered into the equation of Arrhenius, and the mole fraction term calculated. At the end of the rotation exercise, the mole fraction terms were divided by the total of such terms and thus normalized to a sum of 1.0. The matrix (12x 25) of normalized mole fractions is called the population pattern for the given ketone.

Figure 3, page 5, shows a population pattern. For a molecule with one ketone group and one phenyl ring, the pattern is a two-dimensional matrix of numbers. (In fig. 3, the numbers have been represented by various symbols). These numbers represent the percentage of time that the molecule finds itself in the angular position indicated by the choice of abscissa and ordinate. After a substantial number of such patterns was prepared, there was experimentally sought a single matrix which could be used to multiply, term by term, the molecular matrices so as to obtain totals indicative of the spectral properties of the molecules. This latter matrix was called the intensity matrix. The best such intensity matrix found thus far is a simple function of trigonometric values.

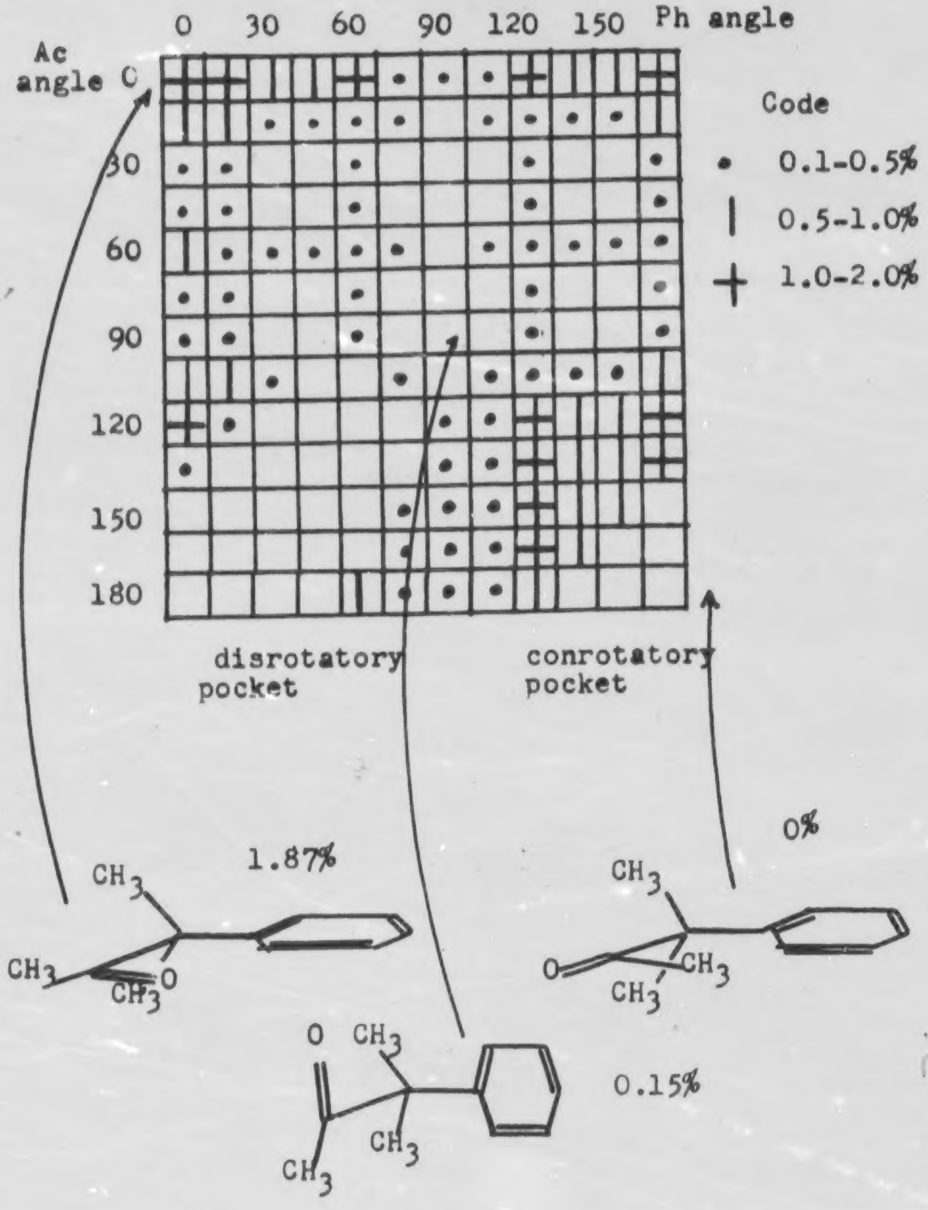
$$I(\psi,\theta) = \sin^{2}\psi \sin^{2}\theta$$

This intensity matrix has theoretical relevance. It is related to the sigma component of orbital overlap of the ketone and benzene groups. Figure 4 shows the plot obtained using this function.

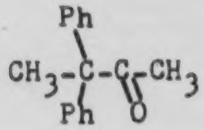
In our later programs, more economical presentation was necessary because of the large number of atoms and rotations. The program for 3,3-diphenyl-2-butanone, for

Figure 3

A Population Pattern



example, gave a population pattern (12x12x25) which was three-dimensional. Calculation of all of the atomic coordinates in such a repetitive manner would have involved a prohibitive amount of computation time. Hence, atomic coordinates were replaced by precalculated repulsion energies, presented as data as a function of angles of rotation.



$CH_3-C(Ph)_2-C(=O)-CH_3$

interaction	data matrix (size)
Ph,Ac	12x25
Ph,Ph	12x12
Me,Ac	25
Me,Ph	12

The program for triphenylacetone required only the first two matrices, the selection of repulsion values being taken through 12x12x12x25 angular positions.

The final result of the preparation of population patterns and multiplication by intensity matrix is shown in figure 4, page 7. On the plot, two lines are shown, both of which are steep functions of the angles of rotation. The upper line, on which seem to fall points for the more highly alkylated ketones, follows the equation below.

$$\epsilon(C_6H_{12}) = 610 \sin^2\psi \sin^2\theta + 20$$

The lower line, the reality of which remains under examination, seems appropriate for ketones less highly alkylated. Change of solvent to EtOH magnifies the difference between the two lines because of unequal solvent response of the ketones. Thus, this computer survey of phenylketones suggested abandonment of the idea of Labhart and Wagniere, a later theory, and reaffirmation of the idea of Cookson, the earlier theory.

Olefinic ketones also should show exalted n-π* transitions and, indeed, the effect has been often noted. The exaltations, however, are much lower than for corresponding phenylketones. This fact, and a number of structural considerations, made the undertaking of a rotation program unattractive. Instead, there will be presented here certain observations

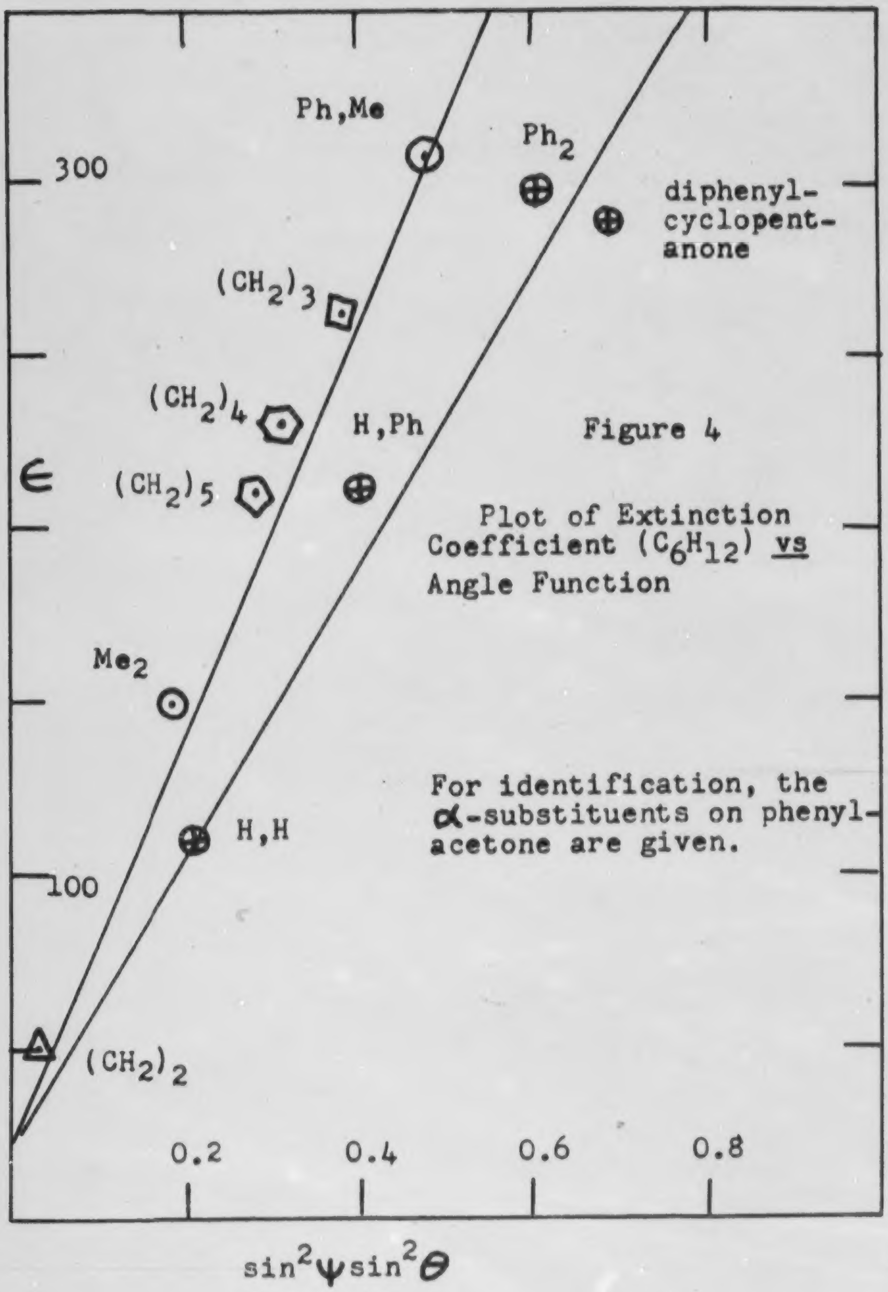


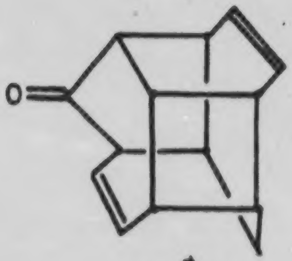
Figure 4

Plot of Extinction Coefficient (C_6H_{12}) vs Angle Function

For identification, the α-substituents on phenylacetone are given.

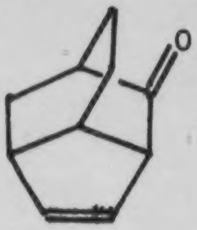
related to molecules of fixed geometry and one theoretical treatment.

If the results of the phenylketone program above have led in the right direction, the maximum value of ϵ of β,γ-unsaturated ketones should be found when the angles ψ and θ are each 90°. Dreiding models of complex molecules VI and VII suggest angles of 90°. The values of ϵ are the largest thus far seen.



VI[8]

301 mμ (220)

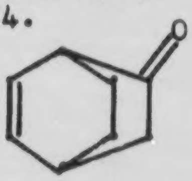


VII[9]

297 mμ (234)

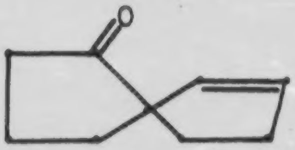
If the extinction coefficients of these ketones continue to remain the largest found, then it can be concluded that β,γ-unsaturated ketones have values of ϵ about 1/4 those of corresponding α-phenylketones.

If the $\sin^2\psi\sin^2\theta$ relationship is accurate, the three ketones below should have the same extinction coefficients and these should be 220 $\sin^2 60\sin^2 60$ or 124.



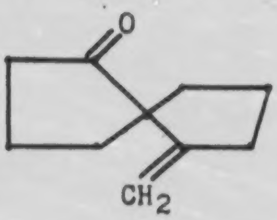
296 mμ (122)[10]

120° 60°



300 mμ (98)[11]

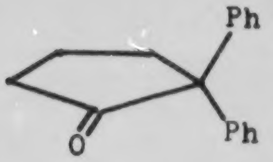
60° 240°



293 mμ (ca 66)[11]

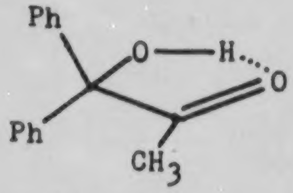
60° 60°

This prediction is seen to be true only in the first approximation and one must ask the question as to whether or not the relationship does not predict values too large at small values of ψ, the angle of rotation of the ketone group. Perhaps the answer is affirmative, for comparison of III and V shows the same effect. Unfortunately, the last compound shown (60°,60°), very sensitive and easily rearranged endo, has not yet been obtained completely homogeneous. The possible failure of the relationship at such small values of ψ may indeed bear on the previous problem, the existence of two lines in figure 4. Certainly for triphenylacetone it can be correctly said that the oxygen of the ketone group must lie near at least one phenyl ring. In dialkylated phenylacetones, however, population patterns showed that the oxygen atom was more likely to be tipped away from the phenyl ring. Hence the existence of two lines in figure 4 may reflect such a difference. Just one more comparison is shown below.



$\psi = 60^\circ$

305 mμ (301)



$\psi = 120^\circ$

284 mμ (468)

This comparison is far from an ideal one, being made complicated by structural uncertainties and solvent differences. Once again, however, the ketone having the greater angle ψ has the higher extinction coefficient.

Olefinic ketones of fixed geometry have data, in summary, which suggest the need for minor revision of the function $I(\psi, \theta)$. The modification may need decreased values when ψ is less than 90°.

The quantum mechanical development of a theory to match this circumstance has not yet been set forth. Our quantum mechanical treatment, which followed the idea of Labhart and Wagniere, is set forth here for the first time. It is readily seen that predicted spectral values are not in accord with observations.

Predicted Spectral Properties of
a β,γ-Unsaturated Ketone

angles		CT band		ketone band	
0	0	216	41	283	0
15	15	212	108	291	18
30	30	211	256	290	68
45	45	212	375	284	74
60	60	213	938	281	102
75	75	215	2186	282	128
90	90	217	3112	285	77
105	105	216	2983	286	27
120	120	213	1795	284	6

Intensities for the CT band are 10^5f and those for the ketone band are 10^6f.

References

1) W.D. Kumler et.al., J. Am. Chem. Soc., 72, 1463, 4558 (1950).
2) R.C. Cookson and N.S. Wariyar, J. Chem. Soc., 2302 (1956).
3) H. Labhart and G. Wagniere, Helv. Chim. Acta., 42, 2219 (1959).
4) W. Baker et. al., J. Chem. Soc., 3357 (1951).
5) D.J. Cram and R.C. Hegeson, J. Am. Chem. Soc., 88, 3515 (1966)
6) K. Takeda et. al., Tetrahedron, 26, 1435 (1970)
7) J.B. Hendrickson, J. Am. Chem. Soc., 83, 4537 (1961).
8) S. Ito et. al., Tetrahedron Letters, 12, 1059 (1967).
9) L.W. Chow and N.L. Allinger, Tetrahedron, 26, 3717 (1970).
10) A. Moscowitz et. al., J. Am. Chem. Soc., 84, 1945 (1962).
11) T.J. Dooley, Ph.D. thesis, Univ. of R.I., 1967.
12) Other references can be found in J. Org. Chem., 35, 2673 (1970).

Data Acquisition and Equipment Control in Microwave Spectroscopy Using a Dedicated Digital Computer

M. D. Harmony, D. K. Hendricksen, P. M. Fast, and D. L. Nordlund
Department of Chemistry, University of Kansas
Lawrence, Kansas 66044

Abstract. The University of Kansas is developing, under a National Science Foundation grant, an automatic data acquisition and equipment control system for experiments in microwave spectroscopy. The heart of the system is a small general purpose digital computer. Programs have been written at the University which enable this computer to operate a microwave spectrometer automatically. Signal averaging and smoothing techniques are used to enhance weak signals. The system is currently used to study the spectra from naturally occurring isotopic species of small molecules; proposed studies include structure and energetics of free-radical species with millisecond life times, generated by a pulsed UV flash or electrical discharge.

The research program in microwave spectroscopy at the University of Kansas is varied and includes the usual molecular structure and dipole moment determinations, measurements of quadrupole coupling constants and barriers to internal motions, and also the study of radicals and other unstable species. The microwave spectrometer is of standard design, with a phase-locked Gertsch system which controls the klystron during computer-assisted operation.

Under a grant from the National Science Foundation, two Hewlett-Packard 2116B general-purpose digital computers and associated equipment were purchased in 1969, one for the microwave spectroscopy laboratory, the other for the high temperature laboratory, in the Department of Chemistry at the University of Kansas.

This paper describes the data acquisition and equipment control system in the microwave spectroscopy laboratory and its present and proposed uses in determining the structure and energetics of various molecules. Presented elsewhere in this conference are papers which report work performed in the high temperature laboratory on computer control of a quadrupole mass spectrometer (1) and a high temperature magnetic mass spectrometer (2).

Figure 1 is a block diagram of the data acquisition and equipment control system presently used in the microwave spectroscopy laboratory. At the heart of the system is a Hewlett-Packard 2116B general-purpose digital computer. This is a small, but fast, stored-program computer, used for real-time, on-line interaction with the experiments in the laboratory. It has a memory cycle time of 1.6 microseconds, 8,192 16-bit words of core memory, and 16 input/output channels in a priority interrupt structure, through which all communication with external devices is accomplished.

Various peripheral devices are connected to the computer to enable it to carry out the operations associated with five aspects of laboratory automation: data acquisition, equipment control, operator interaction, information storage and retrieval, and data processing.

Three devices are used for data acquisition. A high-speed analog-to-digital converter preceded by a data amplifier is used to acquire the customary lock-in detector output voltage from the microwave spectrometer. A frequency counter is used in determining the microwave frequency, and a time base generator provides periodic signals to the computer for various timing operations.

The principal control function is provided by a pulsed stepper motor which is driven by digital signals from the computer and which controls the microwave frequency scan.

Operator interaction is provided by a teletype, an oscilloscope, and a paper tape reader. The operator initiates an experiment by loading the required programs into the computer memory from magnetic tape or paper tape and providing initialization parameters from the teletype. He may later request a change in procedure or enter new parameters by setting appropriate console switches. Parameters may be entered through the paper tape reader or teletype.

Data are presented to the operator either in printed form on the teletype or in graphic form on the oscilloscope or strip chart recorder.

The system has three methods for storing information in a retrievable fashion: directly in computer memory, on the magnetic tape, or via dataphone to another computer system. This may be either the other HP 2116B located in the high temperature chemistry laboratory or the central Honeywell 635 computer.

UNIVERSITY OF KANSAS
DEPARTMENT OF CHEMISTRY
MICROWAVE SPECTROSCOPY LABORATORY
DATA ACQUISITION AND EQUIPMENT CONTROL SYSTEM

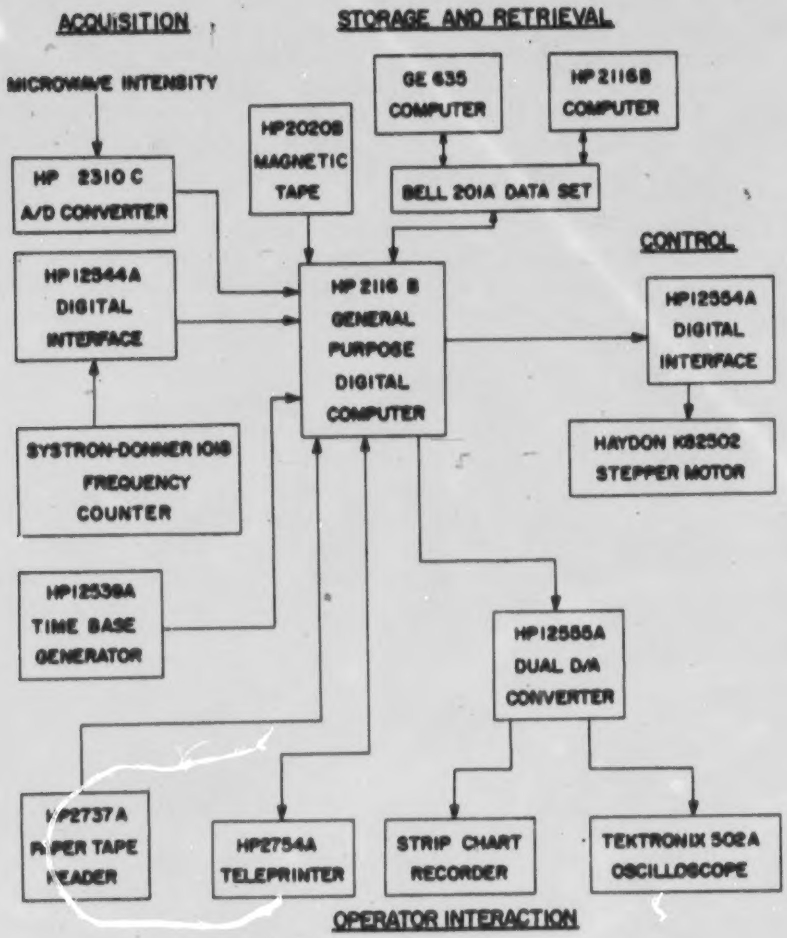

ACQUISITION
STORAGE AND RETRIEVAL
MICROWAVE INTENSITY
HP 2310 C A/D CONVERTER
HP2020B MAGNETIC TAPE
GE 635 COMPUTER
HP 2116B COMPUTER
BELL 201A DATA SET
CONTROL
HP 12544A DIGITAL INTERFACE
HP 2116 B GENERAL PURPOSE DIGITAL COMPUTER
HP12554A DIGITAL INTERFACE
HAYDON K82502 STEPPER MOTOR
SYSTRON-DONNER 1018 FREQUENCY COUNTER
HP12539A TIME BASE GENERATOR
HP12555A DUAL D/A CONVERTER
HP2737A PAPER TAPE READER
HP2754A TELEPRINTER
STRIP CHART RECORDER
TEKTRONIX 502A OSCILLOSCOPE
OPERATOR INTERACTION

Figure I

Data processing includes error checking, converting, scaling, smoothing, averaging, and other data reducing operations. This processing can be performed during an experiment ("real-time") or subsequently from stored data; it can be performed on the local computer ("on-line") or on a remote computer. Timing considerations and memory limitations require that very little processing of data be done in the real-time, on-line mode. Lack of floating-point hardware and high-speed peripheral devices on the local computer requires that any large-scale processing of experimental results be done on the remote central computer.

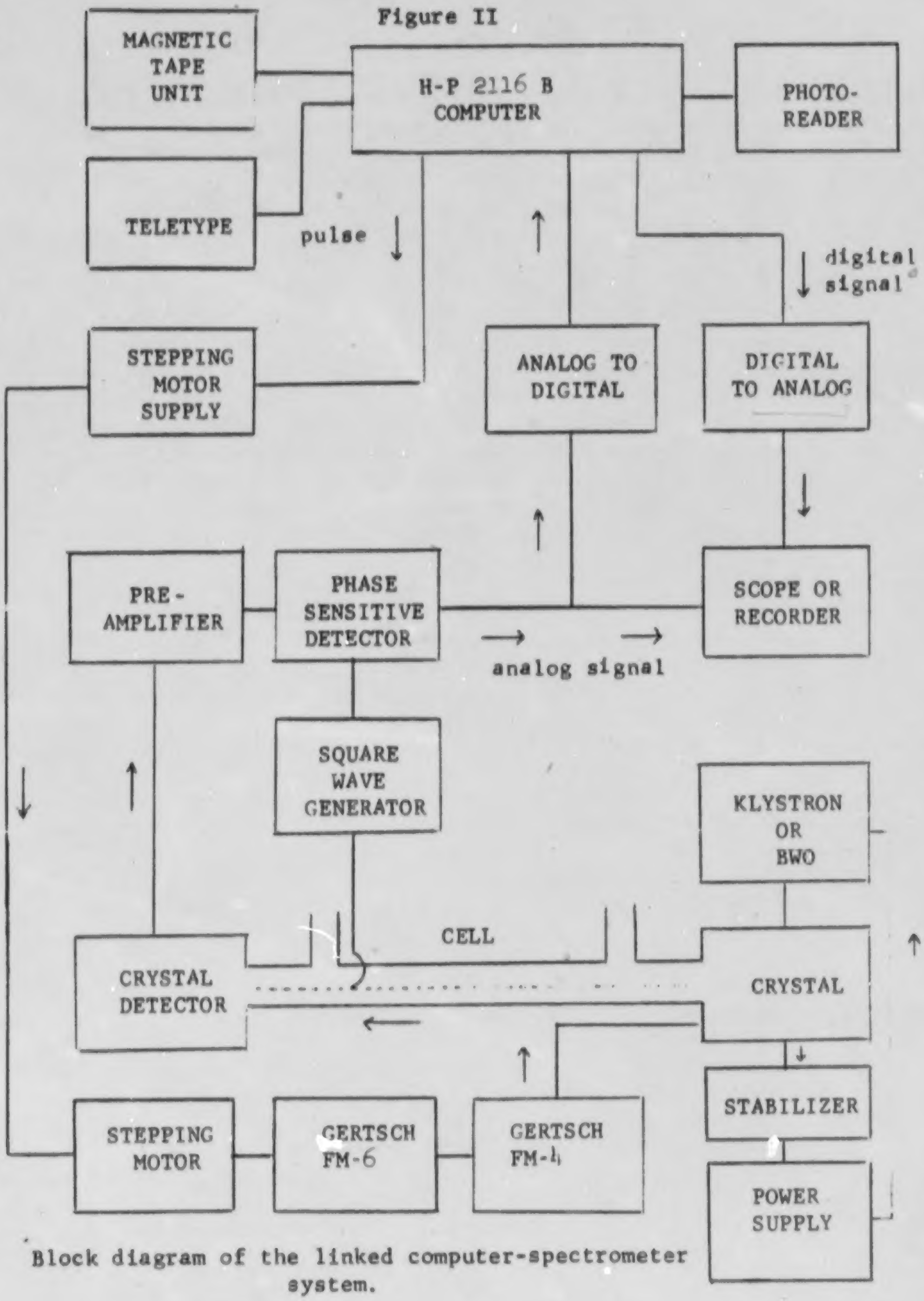
Input/output operations are carried out via a hardware priority interrupt system, which allows computer processing and various input/output operations to be performed simultaneously. Thus, the computer can overlap acquisition, control, display, storage, and processing to obtain maximum speed and efficiency.

Three general types of computer programs are required for these operations: system, communications and applications. System programs translate, modify and load other programs, provide utility functions, and perform hardware tests. Communications programs provide transfer of information between the computer and its peripheral devices, and applications programs are those which direct the computer during an experiment.

One of our first goals for the computerized operation has been weak signal enhancement. We now have the hardware-software configuration which permits spectral searches to be made over narrow-band frequency regions entirely under control of the data acquisition and equipment control system.

Currently, signal averaging is performed in a stabilized mode using a phase-locked Gertsch system, which permits ease and accuracy in microwave frequency variation. Figure II is a block diagram of the spectrometer-computer system. With the stepping motor, the microwave frequency is varied stepwise through a range of 5-10 MHz. At each step, a selectible number of voltage readings at selectible time intervals are taken. The average of these readings and the reading of the counter monitoring the Gertsch FM-6 are stored in memory. Each sweep through the spectral region subsequent to the first is displayed on the oscilloscope simultaneously with the current average of all sweeps. If an individual run is unacceptable for any reason, the operator may reject that sweep during or after the run.

Figure II


MAGNETIC TAPE UNIT
H-P 2116 B COMPUTER
PHOTO-READER
TELETYPE
pulse
digital signal
STEPPING MOTOR SUPPLY
ANALOG TO DIGITAL
DIGITAL TO ANALOG
PRE-AMPLIFIER
PHASE SENSITIVE DETECTOR
SCOPE OR RECORDER
analog signal
SQUARE WAVE GENERATOR
KLYSTRON OR BWO
CELL
CRYSTAL DETECTOR
CRYSTAL
STABILIZER
STEPPING MOTOR
GERTSCH FM-6
GERTSCH FM-4
POWER SUPPLY

Block diagram of the linked computer-spectrometer system.

The software used to accomplish signal averaging gives the operator flexibility and control before, during, and after a run. Initialization parameters include the frequency step size, the number of readings at each point, the time interval between readings, and the counter time base. During a run, the program checks the position of switches on the computer, which allows control during execution of the program.

With this procedure, we are getting two types of averaging. The first of these is short term and occurs at each step where we take repeated voltage readings, and store only the average. The eliminates relatively high frequency noise. The second method is long term averaging which occurs from repetitive scans over the entire frequency band. This eliminates low frequency noise. We also use a third technique, digital smoothing, for increasing signal-to-noise ratio. This is a least-squares procedure which must be used with some caution because it will smooth anything present. Thus a noise spike may appear to be a good absorption line.

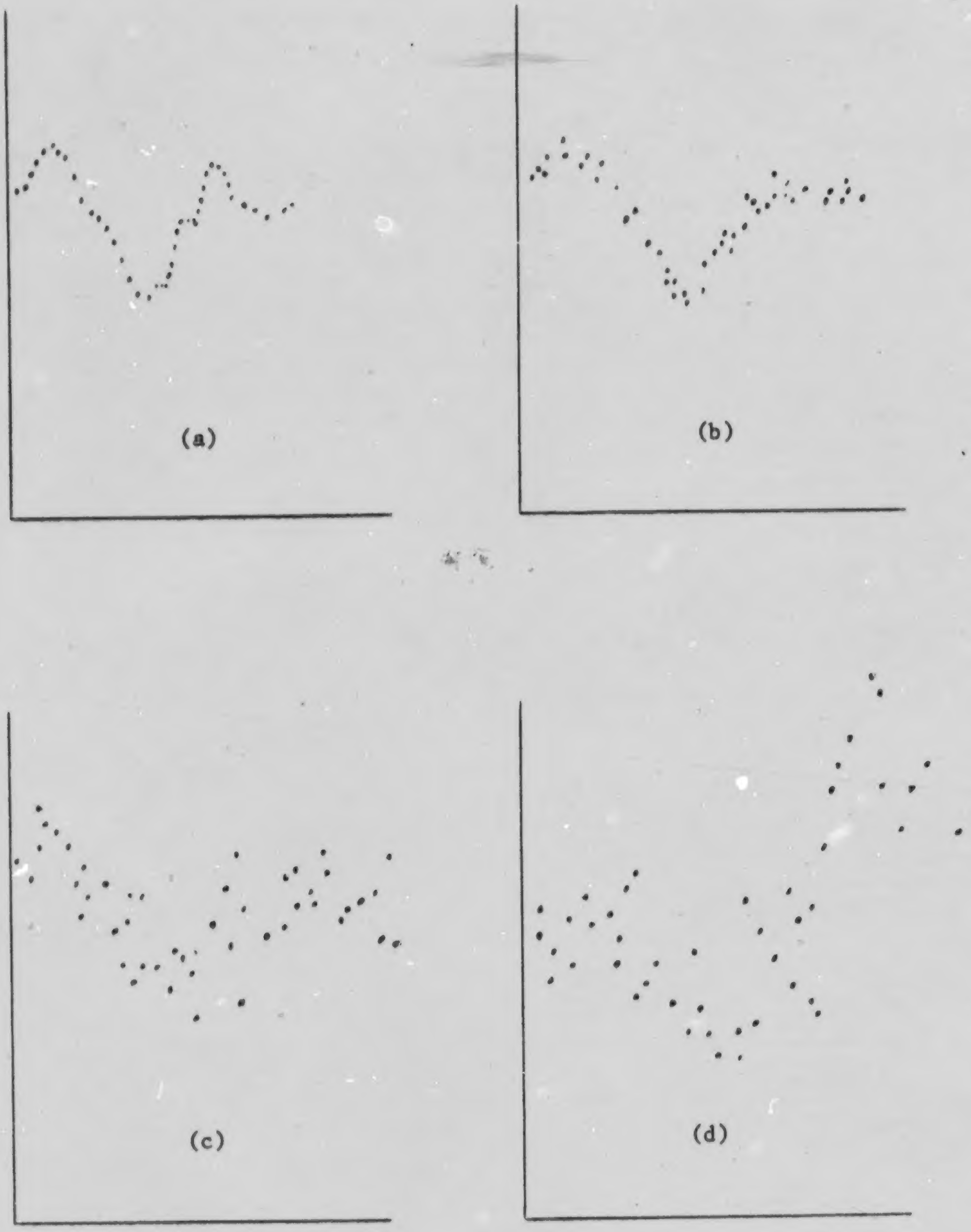
Figure III illustrates the effects of these types of averaging. Part (d) is one sweep through a line, one reading per point. Part (c) is 1 sweep through the line, ten readings per point. Part (b) is 4 sweeps through the line, ten readings per point. Part (a) is 4 sweeps through the line, ten readings per point, smoothed. All other parameters were kept constant. This absorption peak is the ground state J = 1 → J = 2 transition for carbonyl sulfide $O^{18}C^{12}S^{34}$ in natural abundance. The result in (a) could be improved further with increased averaging but it serves to illustrate the procedures.

The usefulness of these signal enhancement techniques (with sensitivity gains of 5-10) is clear. They permit the observation of weak absorption signals which have been undetectable previously. The method finds its greatest use in locating lines of lo[illegible]abundance isotopic species in their natural abundances.

The first molecule we are studying in this way is cyclopropylamine. We are searching for the C^{13} isotopic species in order to determine a more accurate ring structure.

Our developments up to the present have been relatively simple and straight-forward. We believe that with the basic system now operating, it is possible to devise new experiments not previously feasible.

Figure III



For example, with appropriate software developments it should be possible to carry out search and analysis routines for certain spectral features in a real- or nearly real-time sense. This procedure might be carried out as follows. Assume we were interested in locating triplets having 1:5:3 relative intensities. A sweep would be initiated and would be permitted to continue until any absorption feature were detected. An absorption feature would be defined as any signal having a signal-to-noise (S/N) ratio greater than some predetermined value. This signal would be saved and the sweep mode terminated temporarily. The computer-stored signal would then be processed to determine the number of maxima and their relative intensities. Note that some type of frequency width specification for the triplet spacing would need to be made also. If the observed feature conformed within predetermined limits to the desired 1:5:3 multiplet it would be stored for further analysis; otherwise it would be discarded and the sweep mode resumed. Storage of good data could be accomplished on the local magnetic tape or by transfer via the dataphone to the GE-635 system. The latter mode would be most appropriate if large scale processing were needed, or if a nearly real-time analysis were desired.

Another major aim is further development of the data acquisition and equipment control system to make possible detection of the microwave spectra of unstable molecules (or free radicals). The goal here is to study the structure and energetics of species having lifetimes as short as 1 millisecond, and to elucidate their kinetic behavior.

The method to be developed involves the production of high radical concentrations by a pulsed UV flash or electrical discharge, followed by rapid measurements (at rates of up to ca. 30 KHz) of signal intensity as a function of frequency for short periods of time (1 millisecond to 1 second typically). The entire experiment will be carried out under control of the computer. All of the signal enhancement methods mentioned earlier will be utilized.

Other experiments which utilize the rapid measurement technique will be investigated also. These include a broad class known as relaxation phenomena. For example, the time dependence of absorption signals during the on-off square wave modulation period (modulation frequency is 100-5000 Hz) provides information about the transfer of rotational energy between rotationally excited molecules and other (non-excited) molecules.

In these rapid-measurement experiments the principle design problems lie in the timing requirements for the discharge or flash pulse and the subsequent measuring cycle; and also in the handling of the large volume of data which will be generated. Our system has the capabilities for data inputs up to a rate approaching 30 KHz (limited by A-D converter) but, of course, the limited core memory prohibits such a rate for long periods of time. The procedure will be to take data in short bursts and to dump onto magnetic tape files for processing off-line at a later time.

Acknowledgement. The authors are pleased to acknowledge the support of the National Science Foundation under its grant number GJ-332 which provided the computer, the associated data acquisition devices, and the personnel for hardware and software development, and under its grant number GP-15127 for the other equipment and personnel.

References

1. P. M. Fast, J. R. Castles, and P. W. Gilles. "Computer Control of a High Temperature Quadrupole Mass Spectrometer." Paper submitted to Conference on Computers in Chemical Education and Research, DeKalb, Illinois (1971).

2. B. R. Conard, P. M. Fast, D. L. Nordlund, and P. W. Gilles. "Computer Collection of Mass Spectrometric Ionization Efficiency Curves." Paper submitted to Conference on Computers in Chemical Education and Research, DeKalb, Illinois (1971).

AN ON-LINE, TIME-SHARED COMPUTER SYSTEM FOR MOLECULAR STRUCTURE AND DYNAMICS*

John F. Cornelius and Kent R. Wilson[†]

Department of Chemistry, University of California, San Diego
La Jolla, California 92037

I. INTRODUCTION

We will try, in this brief paper, to present those highlights of our two and one-half years experience in using an on-line computer, time-shared among several users, which we hope will be useful to other chemists who have set up or are planning to set up similar systems. First we will describe the types of uses in which we're involved, to orient you as chemists. Second we'll describe our IBM 1800 based system, its hardware and software, of course, but more important the philosophy and approach which have made this an extraordinarily successful venture. Last we'll describe the sort of ideal system we'd shoot for if we were making our plans today, instead of in 1968.

II. CHEMICAL USES

As in any chemistry department, we have a range of faculty interest in computers, from those who find them indispensible to those who find them useless. Four faculty members who felt their research strongly demanded on-line capability, both to run their experiments and to allow "hands-on" types of modelling and data analysis, joined together to write the proposals for the grants which support the system, first to NSF for a Departmental Equipment Grant, and then to NIH for operating and expansion funds. Several types of experiments now run or will run simultaneously on the system (see Fig. 1).

Jim Arnold runs a multiparameter particle detection system for cosmochemical studies.

Murray Goodman, who has just joined the department, plans to interface two Cary spectrometers, one for circular dichroism measurements, through a minicomputer to the IBM 1800 to facilitate his studies of the stereochemistry of polymers, biopolymers, and their model compounds.

Joe Kraut and Nguyen Xuong use the system for X-ray crystallographic determinations of the structure and function of proteins. The X-ray diffraction data is gathered in two ways, first through a diffractometer interfaced to a dedicated PDP-8, which in turn is interfaced to the IBM 1800, and second through a camera and an automatic scanning film digitizer which is directly interfaced to the 1800.

Bob Vold is finishing the hardware to interface his pulsed NMR spectrometer which he is using to study liquid crystals, protein-substrate interactions, and protein denaturation.

Kent Wilson runs his photofragment spectrometer[2] through the 1800 and collects and analyzes several million bits of information per day in studies of dissociative molecular states and of air pollution. In addition, he uses the computer system for on-line generation of computer animated films.

Nguyen Xuong is using the directly interfaced film scanner for human chromosome analysis, as well as for X-ray diffraction studies.

In addition, the above research groups and many others in the Chemistry Department use the system extensively for interactive data analysis, model testing, and course grading.

III. PRESENT SYSTEM

A. Hardware

The system we use, based on an IBM 1800, is shown in Fig. 1. At present we have 32K of 2 μsec core, 30 million bytes of medium speed disk, magnetic tape, two typewriters, card reader and punch, line printer, incremental plotter, several storage CRT's, a mouse, various high and low speed digital inputs and outputs, DAC's, ADC's, and interfaces to film scanner, TV camera and experiments. The purchase cost of the system would be approximately a third of a million dollars, but by buying various subsections as they became available from other computers (using purchase credits built up from rentals paid on other U.C. computers), by trading interfacing for equipment with manufacturers, and by purchasing used equipment, we've assembled the system for considerably less.

B. Software

The system runs under MPX, a multiprogramming executive. Various users have different partitions in core. Most data collection and all data manipulation and storage is by Fortran routines brought in from disk. Core swapping requires less than a second, in most cases, and this is sufficiently fast for virtually all applications. With swapping we can reserve a large background partition for intermediate or final data processing.

C. Philosophy

Our general reason for setting up the system is that each of us can afford more powerful and convenient computer usage by banding together in a time-shared system than we could if we each tried to set up our own separate computer. We feel that demand for usage should flow from below, from the potential users. You can lead a potential user to a computer, but you can't make him use it. A sizable fraction of any group of chemists will find on-line computers useful, but it's only asking for frustration to try to bring everyone into it. Therefore, although we make the computer available to all, we find that only a third of the research groups in the department use it.

We would rather have an old-fashioned system which works than a futuristic one which doesn't. (This doesn't mean we'd buy an 1800 system today -- see the next section.) Within six weeks of delivery we were running data programs and within three months we were time-sharing three experiments. We've learned, particularly in software, to buy realities, not promises.

The economics of a system needs to be tuned for most efficient usage. Anyone can write a program which will tie up a computer almost forever. The hourly charges to users need to be high enough so that everyone thinks about efficiency, but not high enough so that the machine stands idle. Our present rates are $6 per hour. The costs of interfacing new on-line experiments are largely borne by the individual research groups, again to encourage each group to think carefully about the financial realities of their usage.

Staffing is probably the most important decision to be made. One great software or hardware man is worth a half--dozen mediocre ones. A system this size shouldn't be run by a staff operator, but should be open 24 hours a day for operation by the users themselves. They then form a group which quickly educates newcomers. Our system rarely crashes, even with dozens of users.

We've made two serious errors, which still are with us. The first is in having only one permanent staff member. The load is just too great; he can't get away for vacations because so many groups depend on him. The second mistake, and one to which we see no good answer, is that we must write grant proposals several years in advance. The computer equipment we'll buy three years from now is probably only a dream in a designer's head. Thus it's very difficult to justify equipment money in advance, and we are chronically short of it.

IV. IDEAL SYSTEM

The ideal system we'd shoot for, if we were starting in 1971, is shown in Fig. 2. There is nothing original in this concept, but we'd like to add our weight in what we feel is the right direction.

Two things have changed since 1968. First, minicomputers have dropped drastically in price, and it's clear that henceforth almost all new laboratory instruments will come with interface, if not mini, attached. Second, computer centers are becoming more willing, in part because newer computers make it easier, to allow outside computers to tie in. Therefore, we'd try to set up a double or triple layer system, with many minis in the laboratories, perhaps an intermediate processor in the middle, and finally a large computer for number crunching and massive storage at top.

The laboratory end would utilize very small mini computers, with ADC's, DAC's, pulse counters, and other gadgets for digitizing data and controlling experiments. The mini would buffer data between the experiment and the main computer. The terminal devices could be anything with a keyboard and some form of output. Programming of the main computer could be done through the laboratory terminal but more usually would be by remote terminals outside of the laboratory area. The mini computer would interpret instructions sent by the main computer to obtain sequence of operations, number of iterations, output values to the instrument, etc.

The high speed laboratory communications port could be a small or medium computer with its own disk storage and a very limited computational capability. The primary purpose of this computer would be to generate short, fast jobs and act as scheduler and message concentrator for the main computer.

The main computer would be a central data processing facility for a campus or large laboratory. It would probably have multiprogramming capability with a small partition assigned to the laboratory computer system. A high level language would be interfaced to the laboratory communications port. Facilities for generating interpreter strings for the mini would be available.

The sequence of operations would be something like this:

i) The researcher sits down at a terminal and makes connection with the regular time-sharing system running from the central computer. He proceeds to write a program to treat his data. At the point in the program where data is required, he performs an output operation referencing a string of interpreter instructions. This output operation is followed by an input operation referencing the variables he expects to get. Subsequent input operations are controlled by the last interpreter string output. When he completes the program he saves it in the permanent library of the main computer.

ii) When the experimenter is ready to run, the experimenter calls up the program he has written in step i). The interpreter string is transferred to the mini. When the read operation occurs, the communications processor causes workspace to be saved and the mini commences execution of the interpreter string. The data is collected by the mini and sent to the communications processor. The data and control information is saved on the communications processor's disk in a queue of jobs to be run by the main computer. When the communications unit gets to that position in the queue it calls for the saved workspace and passes the data to it. The information is then operated on and any feedback required is sent to the mini. If necessary, a new interpreter string is sent to the mini to change the character of the experiment. If data is to be saved on a file, the main computer does so. The workspace is then freed for another job until more data is available.

iii) After all the data is collected, the researcher goes back to his office and sits down at a terminal. He may now perform any data analysis he wishes. Meanwhile, someone else can be using the laboratory instrument, with the same or different programs in the mini. If more immediate data analysis is required, it can be performed simultaneously with the running of the experiment, and displayed in the laboratory.

The beauty of this scheme is the systems responsibility. There is one central responsible agency; at a college or university probably the campus computer center. Problems such as data management and general analysis programs need to be solved only once. Laboratory programs would be portable from laboratory to laboratory. The additional investment by any one laboratory, once the system was implemented, would be under $10,000, yet direct access would be provided to the power of a large computer.

No systems programming need be done by the researcher. At the very worst he might hire a student programmer to do some of the more annoying work. In addition, a single laboratory mini computer might initially serve several laboratories. The computing center would hire one or two interface consultants and they could provide valuable direction, if not actual hardware, to the researcher at the outset.

Clearly one needs a computer center that is sensitive to the needs of the chemist. With the present financial state of most centers, perhaps they will be more willing to tap a rapidly expanding need than they have been in the past.

The initial investment required to set up a system serving all the laboratory sciences depends on the facilities available at the specific computer center. If there is already a time-sharing system, it should be less than $250,000. This cost is very reasonable, and could be recouped, if necessary, by user charges.

* We thank NSF and NIH for making it all possible.

† Alfred P. Sloan Research Fellow.

1. Ng. H. Xuong, J. Sci. Instr. 2, 485 (1969); Ng. H. Xuong, and S. T. Freer, Acta Cryst. in press.

2. G. E. Busch, J. F. Cornelius, R. T. Mahoney, R. I. Morse, D. W. Schlosser, and K. R. Wilson, Rev. Sci. Instr. 41, 1066 (1970).

FIG. 1. The existing laboratory automation system consists of an IBM 1801, four IBM 2311 disk files, a Kennedy MTS-10/1800 magnetic tape, a Holly line printer, three Tektronix 611 storage displays, an IBM 1442 card reader-punch, and two IBM 1816 selectric typewriters. The PDP-8 has an ASR 35 as IO.

FIG. 2. Our "ideal" laboratory automation system consists of a time-shared central computer, for example a PDP-10, about 100 megabytes of disk storage for remote use and a smaller drum storage for system use. The laboratory communications computer would be a PDP-11, IBM 1130 or 1800, or any computer of that general size. Disk storage would probably be about 1 megabyte. The lab computers would be eight or 16 bit mini computers with about 4K or less storage.

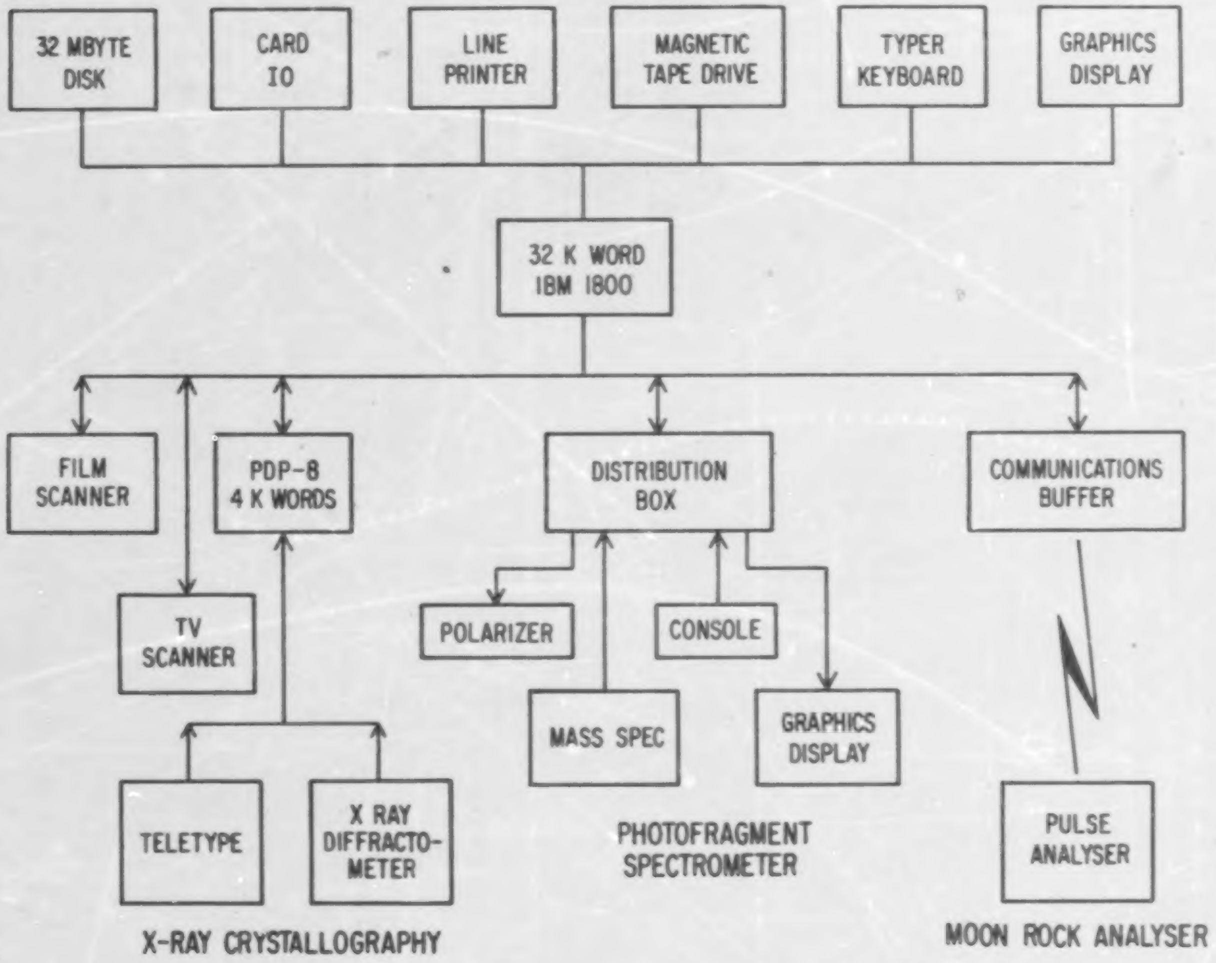

32 MBYTE DISK
CARD IO
LINE PRINTER
MAGNETIC TAPE DRIVE
TYPER KEYBOARD
GRAPHICS DISPLAY
32 K WORD IBM 1800
FILM SCANNER
TV SCANNER
PDP-8 4 K WORDS
DISTRIBUTION BOX
COMMUNICATIONS BUFFER
POLARIZER
CONSOLE
MASS SPEC
GRAPHICS DISPLAY
TELETYPE
X RAY DIFFRACTO-METER
PULSE ANALYSER
PHOTOFRAGMENT SPECTROMETER
X-RAY CRYSTALLOGRAPHY
MOON ROCK ANALYSER

FIG. 1

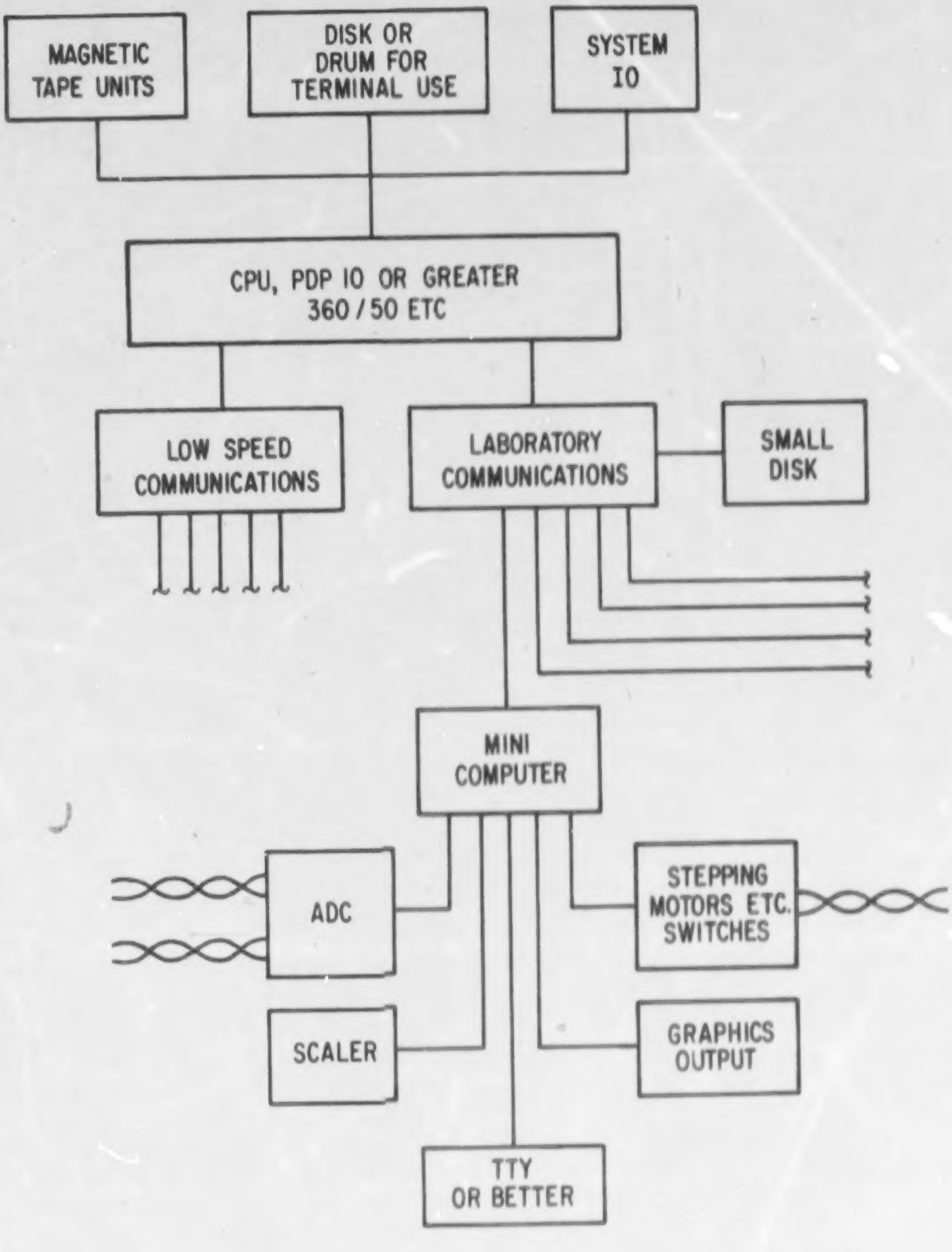

MAGNETIC TAPE UNITS
DISK OR DRUM FOR TERMINAL USE
SYSTEM IO
CPU, PDP 10 OR GREATER 360/50 ETC
LOW SPEED COMMUNICATIONS
LABORATORY COMMUNICATIONS
SMALL DISK
MINI COMPUTER
ADC
STEPPING MOTORS ETC. SWITCHES
SCALER
GRAPHICS OUTPUT
TTY OR BETTER

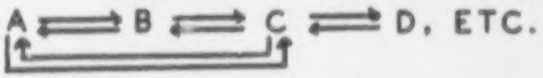
FIG. 2

Application of the Monte Carlo Method to First-Order Rate Equations

John J. Manock and David L. Hooper, Chemistry Department,
Western Carolina University, Cullowhee, N. C. 28723

There have been several articles written which have illustrated the application of the Monte Carlo method to the area of chemical kinetics (1,2). This method is applicable to any system which can be treated statistically as a sequence of purely random events and is particularly useful for systems which do not yield an integrated solution from the closed analytical form or where the physical problem is loss in the complexity of the mathematical solution (3,4). It also eliminates the necessity for a steady-state approximation which can result in erroneous results (3,5).

A general computer program, written in Fortran IV Level F, has been developed which uses the Monte Carlo method to simulate the course for any first-order reaction: reversible, consecutive, and/or parallel. This program is ideal for student use with either a terminal or an on-line printer. The student is required to supply only relative rate constants and initial concentrations, and he can instantly observe reversible, consecutive, and parallel reactions or any combination of these. The flow chart for the program is shown in Figure 1. As in the case of the following general reaction



the non-zero rate constants, R(L,M), and the initial mole fraction concentrations of A, B, C, etc. are provided by the investigator. A matrix is developed with number 1 for molecule A, number 2 for molecule B, etc. These numbers are placed in the matrix in the ratio of the mole fractions. The largest common divider among the rate constants is determined and is designated as TN in Figure 1. This value is chosen as the incremental test factor. Each specific non-zero rate constant is then tested in succession relative to the variable, SN, and if a specific rate constant is larger than or equal to this variable, that molecule can react. Randomly a molecule is chosen from the storage matrix and tested to see if it is the molecule desired. If it is the desired reactant, a fruitful event has occured and this molecule is replaced by the corresponding product.

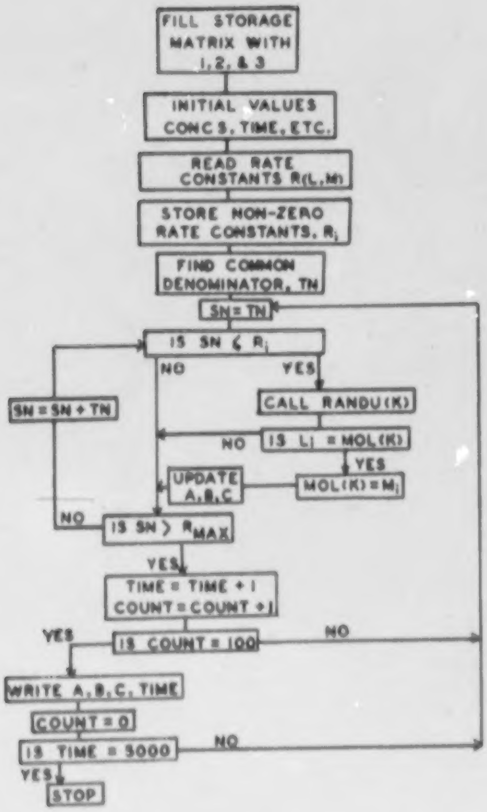

FILL STORAGE MATRIX WITH 1, 2, & 3
INITIAL VALUES CONCS, TIME, ETC.
READ RATE CONSTANTS R(L,M)
STORE NON-ZERO RATE CONSTANTS, R_i
FIND COMMON DENOMINATOR, TN
SN = TN
IS SN ≤ R_i
NO
YES
CALL RANDU(K)
SN = SN + TN
IS L_i = MOL(K)
NO
YES
UPDATE A, B, C
MOL(K) = M_i
NO
IS SN > R_{MAX}
YES
TIME = TIME + 1
COUNT = COUNT + 1
YES
IS COUNT = 100
NO
WRITE A, B, C, TIME
COUNT = 0
IS TIME = 5000
NO
YES
STOP

Figure 1. Flow chart for the Monte Carlo Program.

If the randomly chosen molecule is not the molecule desired, no fruitful event occurs and the next rate constant is considered. When all non-zero rate constants have been tested, SN is incremented by TN and the procedure is repeated. When SN becomes larger than the maximum rate constant, an element of time in noted, SN is initialized to TN, and the process is repeated. At given increments of time, relative concentrations of A, B, C, etc. are tabulated. The reaction is continued until 95% completion has occured or until equilibrium is established. At this point the concentrations of the components as functions of time are plotted on an online printer. The program, including the plot subroutine, requires 50K and takes approximately five minutes to execute on an IBM 360, Model 30, computer. By removing the plot subroutine and reducing the size of the storage matrix, this program can be readily adapted to smaller computers.

To illustrate this program with a specific example, consider the following consecutive reaction

$$A \xrightarrow{1.0} B \xrightarrow{2.0} C$$

where the initial mole fraction of A is 1.0, R(1,2) = 1.0, and R(2,3) = 2.0. The common divider, TN, is 1.0. SN is initially set equal to this value. In the first cycle the rate constants R(1,2) and R(2,3) are tested and the value for SN is increased to 2.0. For the second cycle only R(2,3) is tested, thus, requiring that the molecules A and B be considered in the ratio of one to two. Upon testing R(1,2), if the randomly chosen molecule is 1 then a fruitful event has occured and the storage area is replaced with the number 2, and likewise when testing the rate equation R(2,3), if the randomly chosen molecule is the number 2 then the storage area for this molecule is replaced with the number 3. This process is continued until the reaction shows completion. As shown in Figure 2, results from this program agree within 1% of the integrated values. This precision can be increased by expanding the relative sample size, however, the time required for execution increases proportionally.

This program provides a method for readily evaluating proposed mechanisms of first-order and pseudo-first-order reactions by comparing empirical data with calculated values. A proposed mechanism can often be accepted or rejected by visual inspection. In addition, this program can be used as a method for finding relative rate constants for a complex system by comparing experimental values with calculated values and changing the relative rate constants by a corrective iteration process. The program is currently being applied to problems of this type in photochemical reactions.

The authors would like to express their appreciation to the WCU Computer Center for assistance with this project and the North Carolina Academy of Science for financial support.

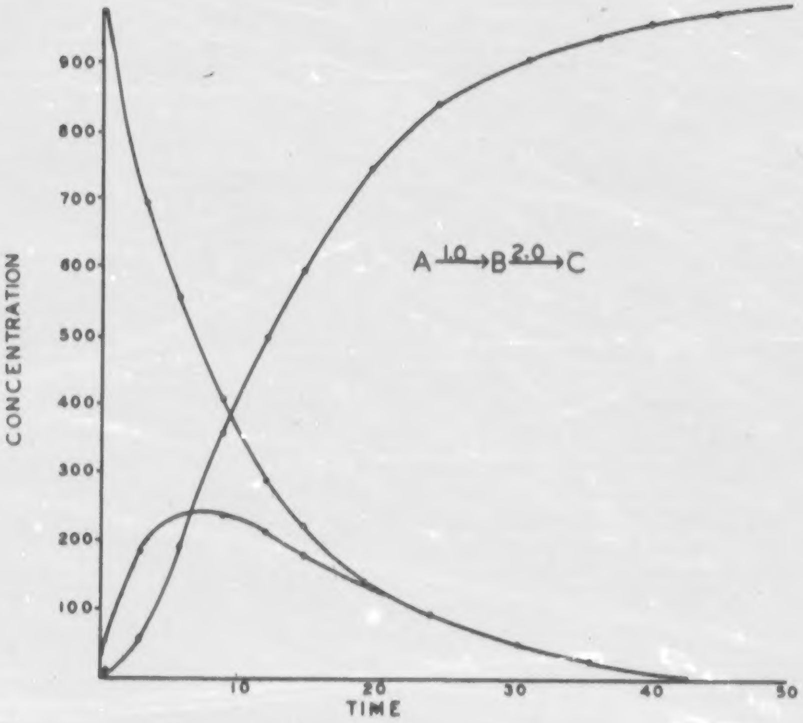


Figure 2. Comparison of integrated values (—) and results obtained from the Monte Carlo method (●).

References cited:

1. B. Rabinovitch, J. Chem. Ed., 45, 262 (1969).
2. L. J. Schaad, J. Am. Chem. Soc., 85, 3588 (1963).
3. E. McLaughlin and R. W. Rogett, Chem Tech, 1, 120 (1971).
4. R. G. Pearson, L. C. King, and S. H. Langer, J. Am. Chem. Soc., 23, 4149 (1951).
5. C. W. Pyun, J. Chem. Ed., 48, 194 (1971).

Series Integration of Consecutive Rate Equations[*]

Howard J. Butcher[+] and Norman C. Peterson[#]

ABSTRACT

The integrated rate laws for two complex reaction schemes are developed in terms of converging series allowing calculation to any desired accuracy. The rate laws do not require tabulated functions for computation. The schemes are:

$$B + C \xrightarrow{k_1} D$$

$$D \xrightarrow{k_2} E$$

and

$$B + C \xrightarrow{k_1} D$$

$$2D \xrightarrow{k_2} E$$

Examples of intermediate concentration [D] vs. time curves evaluated using a computer are illustrated graphically.

---

[*] Taken in part from the Thesis of H.J. Butcher, submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Chemistry to the Polytechnic Institute of Brooklyn, 1966. Work supported in part by a National Institutes of Health Predoctoral Fellowship to H.J.B. and in part by P.H.S. Research Grant No. AM 10946-03 from the National Institute of Arthritic and Metabolic Diseases.

[+] St. Mary's College of Maryland, St. Mary's City, Maryland 20686

[#] Polytechnic Institute of Brooklyn, Brooklyn, New York 11201

Solutions to the differential equations arising from complex, coupled rate equations have been regarded as generally too difficult for exact solutions. In the era before computers were widely available much experimental information was lost by resorting to steady state or other approximate solutions in treating experimental data. Among the many areas stimulated by the increased availability of high speed electronic digital computers, is the non-steady-state intergration of rate equations by numerical methods. Fairly complex kinetic schemes can now be solved by numerical integration algorithms, such as the fourth order Runge-Kutta procedure (4) or predictor-corrector algorithms(5).

Although these are generally useful techniques, questions of accuracy and computation time can arise. In particular, numerical solutions such as these can be in error or even unstable if the integration step size and round off errors are not carefully optimized, and these numerical techniques must be carefully checked against accurate solutions to insure confidence in the results. It is necessary to have exact solutions available in analytical form to check the accuracy of the numerical methods. Integrated rate equations for many of the simpler rate equations of kinetics are known in terms of elementary functions. Solutions for several mechanistic schemes have been obtained in terms of tabulated functions, (6-9) but often such solutions are not readily translatable into a computer program.

In this paper we advocate the use of solutions in terms of infinite series, which are readily programmed for machine computation. Because they are exact solutions they are not subject to numerical instabilities. Series solutions are therefore more accurate than solutions obtained by numerical integration. Finally, series solutions are generally more directly obtainable than solutions in terms of special functions. Few examples are found in the literature of integrated rate laws of any type. It is hoped that examples of this procedure will help to reacquaint chemical kineticists with some classical tools for solving differential equations. The examples presented here are rapidly convergent to high accuracy. These were developed for use in a non-linear estimation program for estimating rate constants and testing mechanisms, a procedure which would have been impractical without the accuracy and speed of a digital computer.

A series solution arising from the irreversible, mixed second order first order sequence is given below. It is shown how to compute the concentration of species to any predetermined accuracy.

The mechanism is:

$$B + C \xrightarrow{k_1} D$$

$$D \xrightarrow{k_2} E$$

The rate equations are:

$$\frac{-dB}{dt} = \frac{-dC}{dt} = k_1 BC \qquad (1)$$

$$\frac{dD}{dt} = k_1 BC - k_2 D \qquad (2)$$

$$\frac{dE}{dt} = k_2 D \qquad (3)$$

Taking Bo and Co as the initial values of B and C, the instantaneous concentrations of the reactants, equation (1) is easily integrated to give the well-known result:

$$\frac{1}{Bo-Co} \ln \left\{ \frac{CoB}{BoC} \right\} = k_1 t \qquad (4)$$

After making the substitution:

$$G = Bo - Co$$

and noting the stoichiometric requirement that B-C = Bo-Co, solving for B and C gives:

$$B = \frac{GBo}{Bo - Co \exp[-Gk_1 t]}$$

and

$$C = B(Co/Bo) \exp(-Gk_1 t).$$

The time rate of change of the intermediate concentration, D, is given by equation (2) or

$$\frac{dD}{dt} + k_2 D = \frac{k_1 G^2 (Co/Bo) \exp(-Gk_1 t)}{[1-(Co/Bo) \exp(-Gk_1 t)]^2} \qquad (2a)$$

Now let $Z = (Co/Bo) \exp(-Gk_1 t)$ and let $\mu = k_2/(Gk_1)$. The differential equation (2a) can then be transformed to:

$$\frac{dD}{dZ} - \frac{\mu D}{Z} = \frac{-G}{(1-Z)^2}$$

This equation has the integrating factor $Z^{-\mu}$.

$$Z^{-\mu}\frac{dD}{dZ} - \mu D Z^{-\mu-1} = \frac{-G}{Z^{\mu}(1-Z)^2}$$

or

$$d(Z^{-\mu}D) = \frac{-G\,dZ}{Z^{\mu}(1-Z)^2}$$

Formally integrating between limits D=0, Z=Co/Bo =Zo at t=0, gives:

$$D = -GZ^{\mu}\int_{Zo}^{Z}\frac{dZ}{Z^{\mu}(1-Z)^2}$$

Define the function R (Z) by

$$R(Z) = \int\frac{dZ}{Z^{\mu}(1-Z)^2}$$

To aid in evaluating the integral, it is useful to apply the binomial expansion

$$(1-Z)^{-2} = \sum_{j=o}^{\infty}(j+1)Z^{j}$$

This series is convergent for $Z^2 < 1$.

Now it can be assumed without loss of generality that Bo>Co and G >0 because of the symmetry of the equations with respect to interchange of B and C. Thus, it is evident that $Z^2 < 1$ for all t. The series is therefore uniformly convergent. After being expanded, R(Z) becomes

$$R(Z) = \int\sum_{j=o}^{\infty}(j+1)Z^{j-\mu}dZ$$

This series is readily integrated term by term. Thus (10) if $\mu \neq j$,

$$R(Z) = Z^{-\mu}\sum_{j=1}^{\infty}\frac{jZ^{j}}{j-\mu}$$

and

$$D = GZ^{\mu}\left[R(Zo) - R(Z)\right] \qquad (5)$$

The concentration of product E, is obtained by difference *viz.* $E = C_o - C - D + E_o$. To examine the convergence of R(Z), the infinite series can be approximated by a finite partial sum with a pre-assigned accuracy. The truncation error, the difference between the nth partial sum and the infinite sum, can be assigned an upper limit (11) by considering the discrete summation variable as a continuous one, so that:

$$\Delta_n < \left|\int_n^\infty \frac{jZ^j dj}{j-\mu}\right| = \left|\int_n^\infty \frac{Z^j dj}{1-\mu/j}\right|$$

where Δ_n is the magnitude of the truncation error in R (Z)/Z.

The denominator can be replaced by $(1-\mu/n)$ preserving the inequality since μ/n is a small positive number less than 1 for j sufficiently large, i.e. $n>j$. Thus an upper limit on the truncation error is given by:

$$\Delta_n < \left|\int_n^\infty \frac{Z^j dj}{(1-\mu/n)}\right| = \frac{Z^n}{(1-\mu/n)\,|\ln Z|} \quad (n>\mu)$$

For example, when $Z=0.5$, $\mu=0.90909....$ the series was evaluated until the nth term was less than 10^{-6} of the nth partial sum. This required 22 terms and led to a truncation error as calculated above, of $\Delta_{22}<10^{-6.4}$ where the sum itself is 6.29286. As the time increases Z becomes smaller, and fewer terms are needed to obtain an accurate value of the sum.

For integer μ the integral can be expressed exactly in closed form (12). Let

$$R_\mu(Z) = \int \frac{dZ}{Z^\mu(1-Z)^2}$$

When $\mu = 1$,

$$R_1(Z) = 1/(1-Z) - \ln\,[(1-Z)/Z] \qquad (6)$$

For $\mu>1$, the recursion formula (7) can be used to reduce the integral (9) to one in terms of $R_1(Z)$, i.e. equation (6).

$$R_m(Z) = \frac{-1}{(m-1)Z^{m-1}(1-Z)} + \frac{m}{m-1}\,R_{m-1}(Z) \qquad (7)$$

The integrated rate equation has been programed as a Fortran subroutine for use with a program for nonlinear estimation by non-linear least squares curve-fitting. (13)

When the ratio of initial concentrations Co/Bo is small the solution for D should reduce to the same form as that for the intermediate in the irreversible first order-first order sequence. This is evident as follows: suppose Co/Bo <<1, then $G \approx Bo$ and $\mu \approx K_2/(Bok_1)$. Only the first term of the infinite series is needed because of the rapid convergence. After substituting in equation (5) and simplifying, one obtains:

$$D = \frac{k_1 BoCo}{Bok_1 - k_2}\,[\exp(-k_2 t) - \exp(-Bok_1 t)],$$

which is the required limiting form.

Calculated curves of the intermediate concentration vs. time are shown in Fig. 1 in which the rate constant k_1 is varied near 1.0. The curve C corresponds to the case in which the closed form solution applies.

The second scheme is the irreversible, mixed second order, second order sequence. A solution to the differential equations of this scheme in terms of Legendre functions was obtained by Kelen (7). However, the series solution presented here can be obtained exactly without need of tabulated function, and is readily programmed for digital computation.

The solution of the differential equations is similar in some respects to the one developed above. Some of the equations are repeated for clarity. The mechanism is:

$$B + C \xrightarrow{k_1} D \tag{8}$$

$$2D \xrightarrow{k_2} E \tag{9}$$

The rate equations are:

$$\frac{-dB}{dt} = \frac{-dC}{dt} = k_1BC \tag{10}$$

$$\frac{dD}{dt} = k_1BC-k_2D^2 \tag{11}$$

$$\frac{dE}{dt} = k_2D^2/2 \tag{12}$$

Taking Bo and Co as the initial values of B and C, the instantaneous concentrations of the reactants, equation (10) is first integrated to give the previous result:

$$\frac{1}{Bo-Co} \ln \left\{\frac{CoB}{BoC}\right\} = k_1t \tag{4}$$

Again making the substitution:

$$G = Bo - Co$$

and noting the stoichiometric requirement that B-C = Bo-Co, solving for B and C gives as before:

$$B = \frac{GBo}{Bo-Co \exp\left[-Gk_1t\right]}$$

and

$$C = B(Co/Bo) \exp \ (-Gk_1t).$$

The time rate of change of the intermediate concentration, D, is given by equation (11) or

$$\frac{dD}{dt} + k_2D^2 = \frac{k_1G^2(Co/Bo)\ \exp\ (-Gk_1t)}{[1-(Co/Bo)\ \ \exp\ (-Gk_1t)]^2} \qquad (11a)$$

Now let $Z = (Co/Bo)\ \exp\ (-Gk_1t)$ and let $\eta = k_2/k_1$. The differential equation (11a) can then be transformed to:

$$\frac{dD}{dZ} - \frac{\eta D^2}{GZ} = \frac{-G}{(1-Z)^2} \qquad (12)$$

This is a Riccatti equation, and may be transformed to a second order linear equation by the change of variable

$$D = \frac{-GZ}{\eta}\ \frac{u'}{u} \qquad (13)$$

resulting in:

$$Z(1-Z)^2\ \frac{d^2u}{dZ^2} + (1-Z)^2\ \frac{du}{dZ} - \eta u = 0 \qquad (14)$$

This equation can be solved directly by the method of Frobenius. First assume a solution of the form

$$u = Z^r \sum_{j=o}^{\infty} a_j\ Z^j \qquad (15)$$

Substituting (15) into the differential equation and equating the coefficient of the first term to zero results in the indicial equation $r^2 = 0$. One solution can then be found from the series expansion

$$u_1 = \sum_{j=o}^{\infty} a_j\ Z^j \qquad (15a)$$

Substituting the expression (15a) into the differential equation (14), and setting the coefficients of Z^j equal to zero gives the following relations among the a_j:

$$a_1 = a_o\ \eta \qquad (16)$$

$$a_{j+1} = [(2j^2 + \eta)\ a_j - (j-1)^2\ a_{j-1}]/(j+1)^2 \qquad (17)$$

The coefficient a_o can be chosen arbitrarily and others can be obtained using the recursion formulas (16) and (17).

A second order differential equation has two linearly independent solutions, and the second solution has the form

$$u_2 = u_1\ \ln Z + \sum_{j=o}^{\infty} b_j\ Z^j \qquad (18)$$

By substituting equation (18) for u in the differential equation and equating coefficients of Z^j to zero, it is found that b_o is arbitrary and the other coefficients are related by the recursion formulas:

$$b_1 = \eta b_o - 2a_1 \tag{19}$$

$$b_{j+1} = [-2(j+1)a_{j+1} + 4ja_j - 2(j-1)a_{j-1} + (2j^2+\eta)b_j - (j-1)^2\ b_{j-1}]/(j+1)^2 \tag{20}$$

The concentration of D is obtained from equation (13):

$$D = \frac{-Gk_1}{k_2}\ \frac{Z\frac{du_1}{dZ} + \lambda Z\frac{du_2}{dZ}}{u_1 + \lambda u_2} \tag{13a}$$

where λ is a constant of integration, and the terms in the numerator are:

$$Z\frac{du_1}{dZ} = \sum_{j=o}^{\infty} a_j jZ^j \tag{21}$$

and

$$Z\frac{du_2}{dZ} = Z\frac{du_1}{dt}\ \ln Z + \sum_{j=o}^{\infty} b_j jZ^j\ + u_1$$

The integration constant λ can be obtained from the initial condition, say at t=0, D=0 and $Z=Z_0$ =Co/Bo. The condition that the numerator in (13a) must be zero at t=0 gives:

$$\lambda = \frac{-\left(\frac{du_1}{dZ}\right)_{Z=Zo}}{\left(\frac{du_2}{dZ}\right)_{Z=Zo}}$$

The concentration of product E, is obtained by difference, viz. E = Eo + (Co-C-D)/2.

The differential equation (14) has regular singularities at Z=0 and Z=1. It can be proved(14) that an expansion about Z=0 has a region of convergence 0<Z<1. Now it can be assumed without loss of generality that Bo<Co and G<0 because of the symmetry of the equations with respect to interchange of B and C. Thus, it is evident that Z<1 for all t. The series is therefore uniformly convergent.

Sample calculations were performed in which the infinite series were truncated when the last term for u'_2 was less than ε, and it was found that the number of terms required decreased with increasing values of t. For example, when Co = 1.0, Bo was 2.0, $k_1 = k_2 = 1.0$ and $\varepsilon = 10^{-9}$, 28 terms were required at t = 0.1 and 18 terms at t = 0.6. The concentrations of intermediate computed were 0.17271636 and 0.53396965 respectively.

Repeating the calculations with $\varepsilon = 10^{-6}$ required 19 and 13 terms with concentrations 0.17271640 and 0.53396835 respectively.

The integrated rate equation has been programed in Fortran and copies of the computer program can be obtained from the authors.

Calculated curves of the intermediate concentration vs. time are shown in Figs. 2 and 3 in which Bo and the rate constant k_2 are varied.

Figure Captions

Fig. 1 Concentration of intermediate, D, *vs*. time, for the mixed second order-first-order mechanism. Co = 1, Bo = 2, k_2 = 1. A: k_1 = 0.8; B: k_1 = 0.9; C: k_1 = 1; D: k_1 = 1.25.

Fig. 2 Concentration of intermediate *vs*. time for the mixed second order- second order sequence Co = 1, $k_1 = k_2 = 1$ A: Bo = 2.0; B: Bo = 3.0; C: Bo = 1.75; D: Bo = 1.25.

Fig. 3 Concentration of intermediate *vs*. time as in Fig. 2 but with Bo = 2.0, k_1 = 1.0; A: k_2 = 1; X: k_2 = 8.0; Y: k_2 = 4.0; Z: k_2 = 2.0; W: k_2 = 0.5.

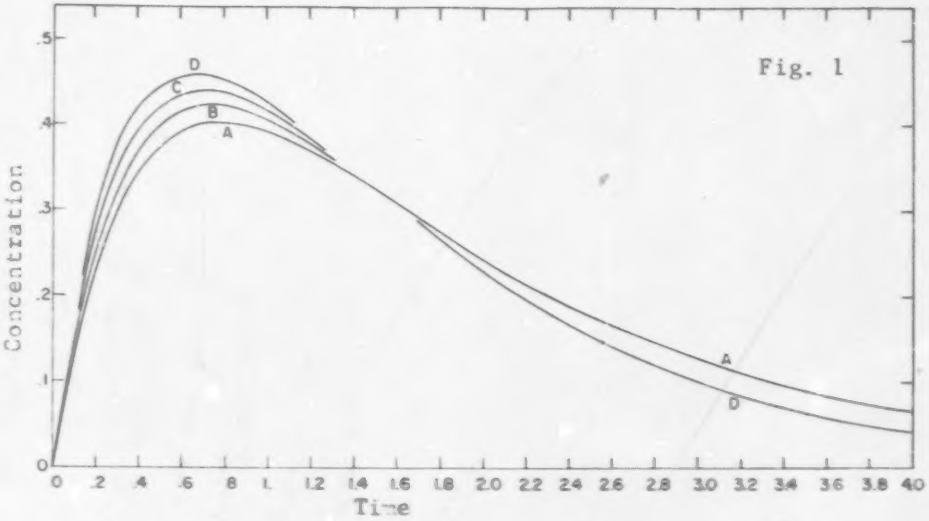


Fig. 1

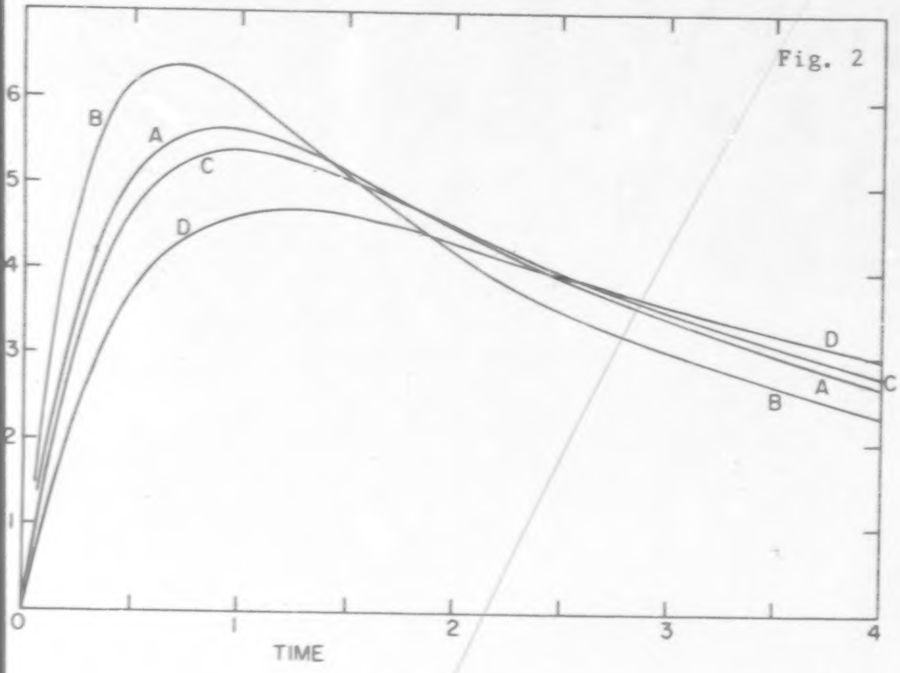


Fig. 2

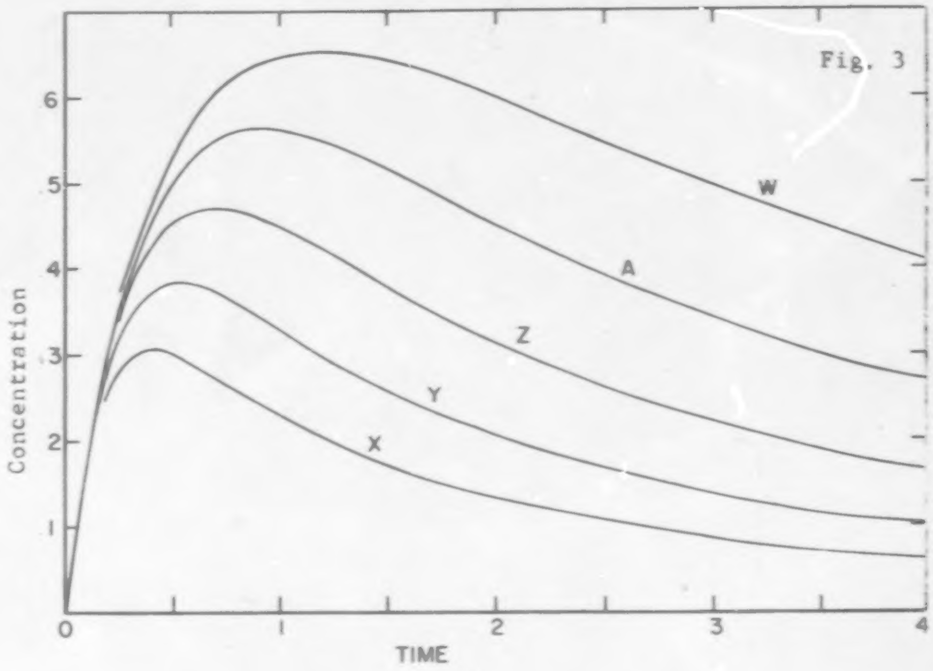


Fig. 3

BIBLIOGRAPHY

1. T. Peterson, Chem. Eng. Process. Sympos. Ser., 56, 111(1960); Chem. Eng. Sci., 17, 203(1962).

2. D. F. Detar, "Use of Computer Techniques in the Study of Reaction Mechanism", Institute of Molecular Biophysics and Department of Chemistry, Florida State University, Bulletin No. 27 J. Chem. Ed. 44, 191(1967); ibid. 44, 193 (1967).

3. D. M. Himmelblau, C. R. Jones and K. B. Bischoff, Ind. and Eng. Chem. Fundamentals, 6, 539(1967).

4. M. J. Kurylo, N. C. Peterson and W. Braun, J. Chem. Phys., 53, 2776(1970).

5. R. W. Hamming, "Numerical Methods for Scientists and Engineers", McGraw-Hill, New York, 1962. Chapters 14, 15.

6. J..Y. Chien, J. Am. Chem Soc., 70, 2256(1948).

7. D. Iltern, Ber. Bunsenges, physik. Chem., 71, 1051(1967).

8. A. S. Baladin and L. S. Leibenson, Doklady Akad Nauk. S.S.S. R., 39, 22(1943).

9. T. Kelen, ibid., 74, 96(1968); Z. phys. Chem. N.F., 60, 191 (1968).

10. After Completing this work an expression for this integral was published by T. Kelen, Magyar Kemiai Folyoirat, 74, 96 (1968); and Z. phys. Chem. N.F., 60, 191(1968), who reduced the integral for non-integer μ to the incomplete beta function.

11. G. B. Thomas, "Calculus and Analytical Geometry", 2nd ed., Addison-Wesley, Reading, Mass., 1958, p. 605.

12. I. S. Gradshteyn and I. M. Ryzhik, "Tables of Integrals, Series, and Products", 4th ed. Academic Press, 1965, p. 59.

13. D. W. Marquardt, J. Soc. Indust. Appl. Math. 11, 431(1963); IBM SHARE distribution No. 309401 (1966).

14. E. L. Ince, Ordinary Differential Equations, Dover, New York, 1956, p. 398

Computers In Chemical Education

by Joseph Lagowski

In the past several years there has been an acceleration of interest in the use of computer-based techniques in the educational process in chemistry. Initial emphasis has been in remedial and/or drill interactions, and recently there has been an interest in mathematical modeling and similation techniques . In fact, there are now available examples of computer methods to supplement virtually every phase of the classical instructional process. Presented in this symposium are papers illustrating many of these points in detail, as well as papers dealing with the broader aspects of using computers such as languages, strategy, results.

The call for papers for this symposium yielded more than three-times the number of papers than the organizing committee anticipated. Accordingly, some rather arbitrary decisions had to be made even after an extra session was accomodated into an already crowded schedule. From, this experience it is apparent that the papers presented here reflect only a part of the present activity in the field.

The Development of Computer Based Instructional Systems in Chemistry

S. J. Castleberry, Dept. of Chemistry
The University of Texas, Austin, Texas, 78712

Computer applications in chemistry have usually taken one of two courses: 1) batch oriented applications such as large number crunching jobs and teaching programming for the solution of chemical problems, of 2) CAI programming of tutorial/remedial drill material. In the first application the most widely used languages are FORTRAN and BASIC; the letter application usually involves a very specialized and highly hardware dependent language and frequently, a dedicated hardware/software system. Examples of CAI languages are:

1. The Coursewriter languages
2. TUTOR
3. Scholar-Teach
4. PICLS
5. CLIC
6. PL-1
7. CATALYST/PIL

Some of the better known dedicated systems are:

1. IBM 1500/1800 Instructional System
2. PLATO
3. H/P 2000B

The two types of applications, batch oriented and CAI, are not mutually exclusive. Indeed, a flexible approach to applying computer techniques to chemical education (and general education) requires that you devise a strategy, without regard to hardware or language, which involves all of the educationally valuable applications of computers. We would suggest that such a flexible strategy should include the following characteristics:

1. the application will supplement the existing course, not replace it.
2. the applications, particularly CAI, will be designed to help students learn, not necessarily teach.
3. the application will help individualize student experiences.

When computer techniques are presented to students as supplemental material they tend to view the computer as an aid and a chance for extra help rather than another dehumanizing barrier between the student and instructor. As a supplement, the computer can remove the burden of stereotyped activities from the teacher and allow him more time for the activities which he performs best.

There are two major aspects to the educational process: the teacher teaching and the student learning. All too frequently teachers become overly involved in helping students learn rather than teaching; That is they have the burden of assigning, grading and giving students feedback on homework and tests; helping students with their assignments; and conducting tutorial/remedial drill group interactions. To a large extent the computer can perform these tasks (on an individual basis) as well as or better than the instructor, for the computer can be programmed to be the world's most patient tutor and has the capability of interacting with the student as often and for as long as the student requires. This, of course, does not diminish the teacher's contact with students but rather, makes it possible for the teacher-student interaction to be richer in the activities which teachers perform best: giving insights into difficult concepts, transmitting an understanding of abstract ideas, inspiring students, and obtaining behavioral objectives in the affective domain.

Computers can be used to individualize student experiences in several ways. CAI programs can be used to measure entering skills and prescribe a series of programs which contain remedial materials if necessary, standard curriculum materials, and/or advanced placement materials. In addition, each program can branch students ahead, provide extra work or help, or branch back on the basis of the students apptitude. CAI programs can allow for individual differences in learning speeds by allowing students to take a module as many times as necessary and work for as long as desired or necessary. When the student's interest dictates, modules which supply specialized or enriched materials can be supplied.

CAI programs can generally be classified as tutorial drill, simulation, or evaluation. With computer based-interactive programs it is possible to provide the student with tutorial drill materials (practice problem sets, question and answer sessions, problem situations, etc) which are tailored to his needs and are unique to him. Simulation programs can be used to extend a student's laboratory experiences to much greater depth than ever before possible, that is, the student can perform a greater variety of experiments and more of each variety and each time collect his own set of unique data. The time compression/expansion capability of the computer allows the student to perform experiments which in the real world occur on a very short time base or a very long time base (ie. kinetic studies). In addition, computer simulations allow students to perform experiments which are too dangerous for beginning students to perform on a large scale in the real laboratory and to perform experiments which are too sophisticated and require too expensive an apparatus for wide scale use by beginning students.

Imbedded within the set of CAI programs can be a series of minimum level examinations which tell the student and instructor that the student has mastered a given concept or task and is ready to move on. On the basis of these minimum level exams the student can be moved ahead, given additional tutorial aid, or the instructor can be alerted that the student is experiencing unusual difficulty or not taking the exams; in either of the latter cases the instructor is provided with the timely information that he should intervene at this critical point in the student's progress. The students can take these exams as they feel prepared and if necessary they can repeat them several times. This is made possible by the computers ability to randomly generate approximately equivalent questions and parameters,thereby making each exam for each student unique.

The computer allows us to "capture" the thoughts and strategies of our most capable people and make them available to large numbers of students. It is then possible for the programs to incorporate rich socratic dialogs, discovery techniques, and response contingent feedback and branching.

Once the general strategy of computer applications has been set comes the task of adapting the available hardware and languages to fit the strategy. For smaller institutions with moderate numbers of students to serve, there is a wide variety of mini-computers and languages which can be adapted to most educational applications. A few of these are:

1. PDP TSS-8 / BASIC
2. H/P 2000B / BASIC
3. WANG 3300 / BASIC
4. PDP - 11 / BASIC
5. Super NOVA / BASIC
6. PDP - 12 / FOCAL

For large institutions which require vast amounts of compute power ther is an even wider variety of time sharing systems and languages. Some of these are:

1. IBM 360/- ,370/- series
   a) APL
   b) FORTRAN
   c) Coursewriter III
   d) BASIC

2. CDC 6000 series
   a) FORTRAN
   b) BASIC
   c) CLIC

3. G.E. 635 (or MARK II r.s.)
   a) BASIC
   b) FORTRAN

4. PDP - 10
   a) BASIC
   b) FORTRAN

5. Others
   a) FORTRAN
   b) Basic

The question of what language or languages to use always arises. On the mini computer the answer is usually BASIC. We have found BASIC to be an excellent all around language for both batch number crunching and CAI programming. One can do with BASIC almost everything you can do with a specialized CAI language and BASIC is a very easy language to learn. APL is a powerful, relatively easy to learn language which is as adaptable to CAI as BASIC but is not as widely available on different hardware system as is BASIC. The other computational languages such as FORTRAN are also ameanable to CAI programming but are more difficult to learn and require personnel with a high level of language expertise.

Specialized CAI languages are usually easy for authors to use but tend to suffer from a lack of computational capability and extreme machine dependee thereby decreasing their value as intact transferrable curriculum materials.

At the University of Texas, the computation center has developed an author language called CLIC. This language has the easy to use operation codes of a course writer type language and the compute power of FORTRAN. The language is in fact written in FORTRAN IV so that curriculum materials may be more easily transferred to other sites with standard FORTRAN IV compilers. When coding in CLIC it is possible to use only the op codes, use only FORTRAN or intermix op code and fortran coding.

We are currently using the CLIC language on the computation center CDC 6600/6400 system and the PDP-10 system to develope a fully computer supplemented introductory chemistry course. The number of formal lectures per week will drop from 3 hours to 1 hour. The student will use these two hours per week for individual and small group interactions with the teacher, tape recorded tutorials and discussions of tests, filmstrips, integrated filmstrip and tape recorded tutorials, CAI programs. The laboratory will consist of a series of integrated real and computer simulated experiments which will attempt to optimize the student's experience with real lab situations and the basic skills of laboratory work and with analyzing and solving experimental problems illustrating basic chemical concepts.

Computer administered and scored tests will be used to obtain a beginning skills profile of each student. On the basis of these results the computer will generate a suggested prescribed course of individual computer and audio-visual materials, suggesting advance placement or remedial work where indicated. The suggested course must of course be approved by the instructor. Once the student begins the course, minimum level exams will be administered via the computer as the student progress. Accurate records are kept which the instructor has access to and can retrive at anytime and on the basis of these records the computer will send timely warnings when the student is not progressing (on the basis of criteria set by the instructor) as he should.

Although the approach we have discussed may sound appealing, many will feel that costs may be prohibitive or that the logistics of getting started may be overwhelming. If you are in a large institution that already has large computational power and/or a time sharing system the major capitol outlay of money is already over the hill and its simply a matter of convincing people of what direction to take.

If one has little or no available compute power, minicomputers offer an opportunity to begin an effort on a small scale at a minimum investment, 25000 to 75000 dollars (actually as low as 1800 dollars for a four teletype, hardware system). The important first step is to devise objectives and set the strategy for developing applications. Then with objectives and stratigies in mind, inventory the available hardware on campus and monies available for support: school in existance, possible grants available via a new proposal. Personnel to help with the project are often one of the easier aspects of getting started. Usually, some graduate assistants can be made available part time, computation centers are often interested in new applications and will make available some manpower. Some of the developmental problems of a new system can provide thesis and dissertation problems for candidates in Computer Science and Science Education. There is also the possibility of acquiring undergraduate assistance either through the work-study program or volunteers.

Computers hold great promise for enriching and adding flexibility to the educational process, indeed much of that promise has already been demonstrated. What is required now is the evolution from demonstration projects to large scale application.

The following is a brief discription of the CLIC System as written by Miss Sally Hartman of the University of Texas Computation Center.

THE CLIC SYSTEM

The purpose of the system, CLIC (Conversational Language for Instruction and Computing), is to incorporate an interactive instructional capability within a large-scale, general-purpose, time-shared computer system (the Control Data Corporation 6600 computer and the UT-2 operating system in use at The University of Texas Computation Center), in a fashion that makes maximum use of the system's resources both in its structure and application.

Approach

In the summer of 1970 a project was begun at The University of Texas to provide users of a general-purpose, time-sharing system with the capability for exploring conversational uses of computers in instruction. The idea for the CLIC project developed from the interest of largely Computer Sciences Department and Computation Center faculty and staff members. They wished to develop the use of instructional conversational programs on the CDC 6600 and to investigate the benefits of CAI in the classroom and laboratory using the existing time-sharing facilities and other routines within the scope of the machine. Support was given by Control Data Corporation for the project development.

The approach taken in the design of CLIC was for the most part a technical one. The design group was composed of computer scientists, not behavioral or information scientists. The group had only faint knowledge of previous CAI endeavors. Local individuals experienced in the field were contacted and their suggestions were incorporated into the CLIC system.

The CLIC system in its initial stages gives to the typical CAI user the capability which he seemed to want on a large, general-purpose computer. This capability mainly consists of a COURSEWRITER-like (question-answer) logic that is easily learned by an author; it's, however, too rigid to be used solely. This serves only as a pivot point, since the CLIC design is extremely modular and the addition of other logics (strategies) can trivially be incorporated. The general (skeleton-like) structure that is provided by the CLIC system, is imperative to a flexible teaching scheme which can truly be dynamic and individualized.

Simplicitic, Flexible Language

As previously mentioned, the CLIC language bears a number of similarities to the well-known, widely-used COURSEWRITER language. Due to this, CLIC appears like a typical CAI system to an inexperienced author, providing him with an extremely simple instructional language.

However, for an experienced user CLIC differs substantially from most previous systems. An author who feels restricted by the CLIC language always possesses the option of dropping into Fortran whenever he wishes.

Significant CLIC Design Features

The fact that CLIC is implemented on a large, general-purpose computer is not solely a significant design feature. The majority of the previous CAI systems that have been designed for and implemented on similar machines, are usually rigid, enormous and completely self-contained instructional systems.

CLIC is not a self-contained system. It is the use that CLIC makes of the machine's potential and the degree to which CLIC is interfaced with the computer, that identifies the important characteristics of its design, and proposes to provide a powerful, integrated and flexible CAI capability to the CDC 6600 computer system.

Every module comprising the CLIC system utilizes the existing software whenever practicable. The major facilities it makes use of are briefly commented on in two parts: (1) an account of CLIC's use of the system's Fortran compiler is covered in the discussion of its "Preprocessor" feature, while (2) the remaining system language processors it uses (i.e., available editing packages and the TAURUS time-sharing and permanent file systems) are sketchily discussed under "Use of Available Software."

(1) Preprocessor

The CLIC system incorporates the feature of a "preprocessor", for the purpose of using a predefined and preoperational computer language and its respective processing system. The preprocessor is the "CLIC Translator" and the language exploited is Fortran.

The CLIC language developed as an augmentation and refinement of the COURSEWRITER language. It was made an extension (a higher level language) to the Fortran language by affixing a "preprocessor" that would translate CLIC source programs to Fortran source programs. CLIC employs the Fortran language as its "basic" system language (i.e., the language of its course programs) and utilizes the Fortran compiler as its system language compiler.

Once a course program has been "compiled" (converted into highly optimized machine code) the course program is ready for student interaction. During a student session the course is "executed" (processed) by the "CLIC Interpreter" (the system's interactive driver).

(2) Use of Available Software

Facilities for course composition and revision, for monitoring telecommunications, and for file manipulation have not been included within the CLIC system. When an author creates his course he has his choice of using any of the available editing routines to build, examine and modify his course program file. When a user converses with a course via CLIC, he utilizes the TAURUS time-sharing system which takes care of all the information transmitted during the interaction. All the necessary file maintenance and file manipulation is automatically accomplished by CLIC via the local Permanent File System.

Evaluation

A general but relatively accurate and comprehensive evaluation of the CLIC design features can be accomplished by listing the advantages and disadvantages of the following two CLIC system characteristics:

(1) Use of a large, general-purpose, time-sharing computer system which takes advantage of the system's capabilities.

Advantages:

a. It makes use of existing software; gives instructional users more extensive software than available to them on dedicated CAI systems.

b. It is more economically feasible than using a dedicated system; before CAI centers spent large sums of money to obtain a dedicated computer or "time-share" a larger system in large blocks of computer time. Their use usually proved costly, since they have to account for (1) their available time, which is usually lost while the computer "idles", waiting for the user to make inputs, and (2) their reserved use of special I/O devices, which as a result of much computer idle time, are not used efficiently.

c. It allows an author to extract previously written routines from other system research users (professors and students, alike) for application in his course. Such common routines, for example, can include those for model building or stimulation, information retrieval, I/O packages for special equipment, etc. In short, course authors are given the potential to relate with others in distinct application areas and to share auxiliary routines.

d. Since CAI time-shares with some of the most technically advanced in the computer field, it is given the opportunity and is shown the need, to refine its interaction.

e. When a CAI system is integrated within a complex operation system and linked to existing software, its capabilities are immediately provided and promptly improved with a minimum investment in system programming and program documentation.

f. It gives the regular user of the system another utility routine that can help him in his work. For example, a user can interrogate CLIC for a quick programmed review of a subject (without leaving his terminal) in order to detect his error when using some system routine.

g. This instructional system adds another application area to a large centralized computer system, which will financially help the computer center in its continual effort to provide overall efficient, reliable computing power.

Disadvantages:

a. The large system does not provide the wide variety of I/O devices, as do most of the smaller machines that are dedicated to CAI and extended specifically for use in education. Equipment such as individual CRT's, slide projectors, recorders, etc., is extra from the actual computing service used by CLIC.
b. The author who writes a course is expected to know a little about the computer's software. He must be knowledgeable in the basics of additional language processors, such as, TAURUS, an editing routine, the Permanent File System, the Fortran language, etc.)

2. Use of a preprocessor to define the basic CLIC system language to be Fortran.

Advantages.

a. Provides the obvious advantage of ease of coding and checkout.
b. Immediately provides full Fortran capability with little programmer effort and extra program documentation; these include computational, sequencing (e.g., GOTO's, conditional and unconditional branching, function and subroutine features), logical, I/O, formatting facilities and access to a large library of functions.
c. Provides that most thorough linkage to routines and packages within the operating system. The CLIC system is automatically given access to the system's program library; system I/O and editing packages, permanent file and file maintenance routines, special packages for data manipulation (SLIP), display routines (plotter, film, CRT, etc.), are immediately available to CLIC users.
d. Fortran subroutine capability makes the CLIC system and its language readily extendible. It allows for easy addition to the set of answer-processing functions that enhance the system, and to the set of auxiliary functions that expands the author language, thus increasing user capability.
e. An author who is restricted by the CLIC language may intersperse actual Fortran code or intermix supplementary routines

with his CLIC section, or he may completely replace a section with all Fortran routines. In short, the scoop of the resulting language for course programs is extremely flexible; the use of Fortran provides an author means of leaving the higher level author language whenever it becomes too restrictive.

f. Compiled Fortran code yields more efficient data for the CLIC driver during interaction, than interpretive data. It provides a more efficient execution (i.e., faster response time) of CLIC programs during an instructional session, since the immediate course section is completely in core and only minimal I/O - that to the student's terminal - is required.
g. Fortran is relatively machine independent. Source code for both the CLIC system and its courses can be implemented on any suitable large, general-purpose computer; courses that are written for one computer can be easily recompiled for another which contains CLIC.
h. The author is able to modify his text and prepare alternative versions with relative ease.

Disadvantages:

a. Fortran is not readily adaptable for string processing; therefore, no real string processing capabilities are present.
b. Preprocessing usually requires more time in a system which compiles machine code that is to be executed, than in a system which does little to course data files that are to be interpreted. Even minor editing of a CLIC section requires re-translation and re-compilation.
c. An author who is restricted by the CLIC language and wants to use the potential of Fortran, must at least be knowledgeable in basic Fortran.
d. In core execution of CLIC programs is expensive in the amount of core required. Systems that interpreter their data files, however, need only that core required by the interactive system since their data is stored on disc.
e. Clumsy I/O is inherent in Fortran; formatting information within an instructional program is not easily accomplished if an author is restricted by the CLIC language and reverts to Fortran.
f. Fortran is not truly machine independent.

THE INTRODUCTION OF DIGITAL COMPUTER TECHNOLOGY INTO THE UNDERGRADUATE CHEMISTRY LABORATORY

by

S. P. Perone and J. F. Eagleston
Department of Chemistry
Purdue University
Lafayette, Indiana 47907

Current trends in the chemical industry throughout the country indicate that the digital computer will soon become an indispensable part of chemical experimentation (1,2,3). It should be clear, then, that the chemist of the near future will be required to understand and use properly digital instrumentation and digital computers in the laboratory. Furthermore, educators at colleges and universities will need to develop undergraduate curricula providing the required background in computerized chemical instrumentation. In fact, the demand for this kind of training by practicing scientists has prompted the development of a summer short course here at Purdue for college teachers and others in chemistry and the biomedical sciences (4).

This paper describes the first steps in a long-range program involving the introduction of on-line computer applications in the undergraduate chemistry laboratory at Purdue. The introduction has been made as part of an introductory analytical chemistry course dealing with quantitative chemical measurements (junior level at Purdue). A primary objective was that none of the chemical and instrumental principles currently developed in the course be eliminated, but rather that these be augmented by the incorporation of the on-line computer into many of the laboratory experiments. The manner in which such a program may be made feasible is described below.

APPROACH

It was our intention to introduce computer technology into the undergraduate chemistry laboratory without slighting the chemistry content. The approach has been to continue to include the same basic set of laboratory experiments--except that, eventually, many will be designed to require on-line communication with the digital computer. The student is required to understand and execute the laboratory experiment as usual--but, in addition, he is required to design the appropriate programming and communication elements for optimum interaction between the computer, experiment, and experimentalist. Thus, we envision the student's achievement to be one of experimental design, and the proper utilization of computing equipment in the solution of measurement problems in chemistry.

Specific aspects emphasized include: the general-purpose nature of the digital computer (i.e., how it can be applied to a large variety of experimental problems with modifications in programming); the computer as a control element in experimentation; the rapid response of the computer, allowing real-time interaction with experiments--with reaction times several orders of magnitude faster than the human operator; and the computer as an integral part of the chemical instrumentation.

To accomplish the above mentioned objectives, it was necessary to recognize, first of all, what could not be expected within the scope of this course: (1) we could not educate our students in the intricacies of machine-language programming, and (2) we could not provide them with the electronics background requisite for the sophisticated design and construction of interfacing between computer and experiment.

What could be done is to take advantage of students' previous exposure to off-line programming (with FORTRAN usually), which many--but not all students--have obtained before the Junior year at Purdue. We could also take advantage of their exposure to the fundamentals of electrical measurements in Physics courses to instill them with the most elementary concepts of amplification, response, and noise. We could expect to provide them with the essential principles of digital logic required for interface design.

The most important requirement for the success of the program, however, was the development of two items: (1) a modified high-level programming language, which includes versatile data acquisition and experimental control subroutines with conversational mode calling sequences. (2) A general-purpose hardware package for interface design. This apparatus would provide for patch-board incorporation of those digital and analog modules required to complete the interface between computer and experiment. The general-purpose interface package requires only that the student be able to lay out the basic logic design, timing sequence, and the analog amplification or attenuation required. He can then select appropriate pre-packaged plug-in units to implement his design. He does not have to tend to such details as level conversion, logic conversion, noise rejection, and other subtle design parameters. Yet, he will accomplish the most important part of the interface development--logic design.

PURDUE REAL-TIME BASIC SYSTEM

The high-level programming language chosen for development in this work was the BASIC language (5). There were several reasons for selecting this language: (1) It is easy to learn, generally requiring the order of half a day exposure to develop a good working knowledge. (2) It is an algebraically-oriented conversational language. (3) It is interactive. That is, the compiler is interpretive, and therefore compiles and executes programs line-by-line. This allows programs to be entered, executed, and edited on-line thru a teletype terminal. It also provides for immediate turn-around and rapid error diagnostics. (4) BASIC is rapidly becoming a universally acceptable language. (5) BASIC is currently readily available at commercial time-share terminals. This allows the convenient and economical learning of the language by large numbers of students for later on-line experimental applications.

It should be noted that other languages (*e.g.*, FOCAL*) are available which have comparably desirable characteristics. Even languages such as FORTRAN or ALGOL which are generally considered strictly for off-line applications may be appropriate, particularly for systems with "load-and-go" capability.

The software system developed at Purdue will be referred to as "Purdue Real-Time Basic," (PRTB). The software includes, fundamentally, the BASIC compiler available from Hewlett-Packard Co., which manufactured the computers used in the undergraduate laboratory program (see Experimental Section). The modifications generated here at Purdue have involved the development of a series of machine-language subroutines which are directly callable from the BASIC software, and which are designed to communicate in a variety of ways with experimental systems. The PRTB data acquisition and control software are described below.

General Description of PRTB System.

Figure 1 illustrates the general nature of an on-line computer system. On the one hand is located the laboratory experiment, the data from which the student desires to present to the computer for data processing. On the other hand is the digital computer itself, which is capable of high-speed programmed computational and control functions. In order to operate the communication link between the computer and the experimental system an electronic interface must be established. This interface accomplishes the functions of translation (analog-to-digital, digital-to-analog, decoding, logic conversions, *etc.*), timing, synchronization, and logical control. The programming language must take into account the nature of the electronic interfacing; and the interfacing must be designed with the fundamental characteristics and capabilities of the programming language in mind. The following section describes the subroutines developed for on-line experimentation within the PRTB system. A summary of these subroutines and characteristics is given in Table I.

---

*Trade Mark, Digital Equipment Corp., Maynard, Massachusetts.

Experimental Input/Output Subroutines for PRTB.

Data Acquisition. Two different types of data acquisition subroutines are available within PRTB. They both work in conjunction with an external clock-controlled analog-to-digital converter (ADC). One is a subroutine (SB3) which should be called by the user's program whenever the computer should be waiting for the next data point to become available. When the next datum is digitized, SB3 takes the data point from the data acquisition device (ADC), converts the datum to floating point format, and stores it in the appropriate memory location for subsequent reference by the BASIC program. When SB3 is exited, the computer returns to the next statement in the BASIC program for execution. The new datum can be operated on, but, before the next data point can be acquired, SB3 must be called again.

A second type of data acquisition subroutine (SB7) is one which allows for the acquisition of a complete block of data before exit. The external clock is started and synchronized with data acquisition within the subroutine. The basic difference between SB7 and SB3 is that SB3 is called to acquire one data point at a time and therefore allows for program statements to be executed during the time between acquired data points. Thus, an experimenter could devise a program which could process experimental data while the experiment was in progress and data were being acquired. This is referred to as "real-time" data processing. However, because the computer program must involve the relatively inefficient execution of BASIC statements between data points, there is a more severe limit on the speed with which data can be acquired without the computer getting out of synchronization with the experimental timing. In fact, SB3 is designed to detect when the computer has gotten out of synchronization; an error message will be typed, and the computer will halt. The user will then have to revise his program to require less real-time processing. With SB7, on the other hand, a complete block of data points are acquired before the subroutine is exited and control returned to BASIC. Therefore, the timing is limited by the efficiency of the machine-language programming developed within SB7. For our computer systems and software, it is possible to acquire data at rates as great as 20 KHz with SB7, but no real-time data processing is possible.

The limiting data rate when using SB3 is about 500 Hz, with a minimum of real-time data handling. If several BASIC computations are to be executed between data points when using SB3, the limiting data acquisition rate may be the order of 1 to 10 Hz. However, this is generally more than adequate for most experiments in the chemistry laboratory.

Experimental Control and Logic. In addition to data acquisition, there are several other ways in which the computer can communicate with the experimental system. One of these is through a digital-to-analog converter (DAC), where the digital output of the computer is converted by the DAC to an equivalent analog voltage level. One of the PRTB subroutines (SB9) provides the capability for driving the DAC, the output of which can be connected to external experiments. It is possible to generate a continuous voltage waveform output from the DAC by mathematical generation within BASIC of the discrete points making up the waveform and transmitting these through SB9 to the DAC in a repetitive fashion synchronized with an external clock.

Subroutines SB8 and SB11 allow the programmer to utilize specific output bits on a selected I/O channel to control external devices. There are 16 output bits available for this function. With SB8, the user can select which bit he wants to set by specification of the bit number. The setting of one of these output bits causes a corresponding binary voltage level change at the specified output terminal, and this can be used to close or open switches, start or stop experimental events, light indicator lamps, etc. SB11 allows the programmer to output a 16-bit binary voltage pattern with any simultaneous combination of "1"'s and "0"'s he chooses. This binary pattern is selected by including in the subroutine call the decimal equivalent of the binary number to be generated. Thus, more than one event can be controlled simultaneously.

SB12 and SB13 provide digital input information for program "sensing" of external situations. Thus, 16 binary voltage input terminals are available to the user, the status of which are acquired by either subroutine. SB12 transmits to the BASIC program the numerical equivalent of the input 16-bit binary voltage pattern. Because the status of these bits can be set by external events, the computer could use this information to make appropriate changes in the data processing or control programming.

SB13 is similar to SB12, except that only the status of a single specified input bit is determined. This subroutine is useful to the student because he need only specify a bit number in the call statement to check the status of a bit.

Timing and Synchronization. The most fundamental operation which must be accomplished by the PRTB system is the generation of a time base for all experimental functions. This is accomplished by the incorporation of a fixed frequency crystal clock (10 MHz) into the interface hardware. Also included is electronic countdown logic to scale the output clock pulses down to a usable frequency range for chemical experimentation (100 KHz to 0.01 Hz). The countdown logic can be modified under program control to select any frequency within this which can be generated by a decade-and/or a 1,2,5-countdown sequence. The programmed clock output pulse train is then available to control the timing on the ADC, DAC, or any other external hardware. Also available simultaneously are synchronous clock pulses representing frequencies at the various stages of countdown. (These details are described in the Experimental Section.)

The clock is controlled by the PRTB software through subroutines SB1, SB2, and SB10. SB1 is the initialization subroutine. Through it the programmer specifies the clock frequency, within the limits outlined above. (SB1 is also used to specify the symbol assigned to the variable which will take on the values of the digital data acquired in SB3 or SB7.)

SB2 is called when the computer program decides that it is time to enable the clock. That is, an ENCODE bit is set to a "1" state on the data acquisition channel. This bit can be connected externally with a patchcord to the ENABLE terminal on the clock module (see Figure 2-A). Alternatively, the ENCODE bit may be brought to some external logic, the output of which will set the ENABLE clock input when other external events have occurred--like the start of the experiment. One possible external logic configuration is shown in Figure 2-B, where the ENCODE bit conditions one input of an AND gate. The other gate input is conditioned "TRUE" when the experiment has been initiated. The AND gate output will then enable the clock. At that point the countdown logic will be enabled and programmed clock pulses will begin to appear at the available outputs.

SB10 is called to disable the clock when the time base is no longer required. It simply "clears" the ENCODE bit on the data acquisition channel which has been used externally to enable the clock.

Another type of communication required between the experiment and computer is associated with synchronization between computer operations and external events. For example, if the computer has completed all preliminary program execution required before being able to accept experimental data, the user may choose to receive an output from the computer which may not only tell the experiment to start, but may also initiate external events associated with the experimental system. One subroutine available for this kind of operation is SB5. SB5 causes the output of an ENCODE bit on a selected I/O channel to change state (go to the "TRUE" level) when SB5 is called. Then the computer waits within SB5 for an external event to occur which will cause a FLAG bit on the same I/O channel to change to a TRUE state. The subroutine detects this event and then allows the next sequential program statement to be executed. This next statement might conceivably be a call to SB2 which initiates data acquisition.

A simpler, but less precise, means of communication with the computer for the purpose of synchronization is provided by SB4. When this subroutine is entered the computer simply waits until the operator flips a toggle switch corresponding to bit 0 on the computer console switch register. The computer then exits SB4 and executes the next program statement--which might be to call SB2 and start data acquisition.

Finally, a subroutine (SB6) is available which allows general-purpose timing functions. This routine simply waits for the FLAG bit to be set on the data acquisition channel, clears the FLAG, exits the subroutine, and the next sequential program statement is executed. The purpose of this subroutine is synchronization of program segments with the external time base. (Note that SB5 can be used, also, for general-purpose timing functions if the data acquisition channel is needed for other purposes. The user must simply connect the clock output to the FLAG input of the alternative I/O channel.)

EXPERIMENTAL

Computer Instrumentation.

The digital computer system in this work was a Hewlett-Packard 2115A, equipped with 8K core memory, high speed paper tape input, ASR/33 Teletype, a 16-bit 20 usec DAC, and an interfaced Tektronix Model 601 oscilloscopic display. In addition, the computer has an interfaced data acquisition system and a general-purpose experimental interface capability. (These are described in detail below.) The complete computer system is mounted in a cabinet which has roll-around capability. The software used for laboratory on-line operation (PRTB) utilized the H.P. BASIC compiler, H.P. #201112A. This software is made available from the computer manufacturer. The program listings of the additions to the H.P. BASIC compiler made here to implement PRTB are available from the authors upon request.

Data Acquisition and General-Purpose Interface Hardware.

A schematic diagram of the data acquisition system operated in conjunction with PRTB is shown in Figure 3. The analog-to-digital converter (ADC) used in the data acquisition system was manufactured by Digital Equipment Corporation (DEC), Maynard, Mass., (#C-002, 10-bit, 33 usec conversion time, 0 to -10.23 V. input range).

The programmable clock is constructed from a 10 MHz crystal-controlled oscillator scaled down to usable frequency ranges with MSI (medium-scale integrated circuit) programmable countdown logic. A schematic diagram of this module is given in Figure 4, with a complete list of hardware components. Programmed clock pulses are available from 100 KHz to 0.002Hz. Each decade frequency can be divided by 2 or 5. The specific frequency is selected by a 5-bit output word which is decoded in the clock module by 2 Texas Instrument SN74151 digital multiplexers. In addition to the programmed output, each decade output from 1 MHz to 0.001 Hz is available externally by patchboard connection.

Also available on the data acquisition panel were an ENCODE output bit which could be set and cleared by the computer, and a FLAG input terminal to allow external setting of the FLAG bit on the computer Input/Output (I/O) channel used for data acquisition. The end-of-conversion (EOC) flip-flop of the ADC was normally connected to the FLAG input; however, it was possible to connect any appropriate externally generated signal to the FLAG input. Six bits of digital information could be transferred to or from the computer thru the data acquisition panel using patchboard connections on the panel. Other generally useful functions available on the data acquisition panel included patchboard connection to various logical devices such as 16 AND, OR, and NOT gates, 4 flip-flops, 2 one-shots, 4 analog switches, 4 relay drivers, a Schmitt trigger, a track-and-hold amplifier, 6 indicator lamps, and 2 push button switches with Schmitt trigger outputs. (See Ref. 6 for a general discussion of characteristics of control logic modules.) DEC R-series logic (7) Flip-Chip cards and power supplies were used for logic functions. For interface design which required more sophistication or more logic elements than could be obtained on the data acquisition panel, a DEC patchboard R-series Logic Lab (7) was available. Most experiments did not require interface hardware beyond that available on the data acquisition panel. All input and output to the computer were buffered with level conversion devices on the H.P. 2115A system so that all external connections are compatible with DEC R-series positive logic (7). DEC W601 and W510 level conversion cards were used.

An interfaced digital I/O module was also available for general-purpose digital communication between computer and experiment. The characteristics of this module are illustrated in Figure 5. This digital I/O Module was connected to a different I/O channel than the data acquisition panel. Thus, a separate independent ENCODE output and FLAG input were available thru this module. The use of the digital I/O capability has been described above.

RESULTS AND DISCUSSION

Evaluation of the PRTB System.

For the purpose of illustrating the capabilities of the PRTB software for this publication, the laboratory computer systems and software were tested with two synthetically generated experimental outputs which simulated typical experimental data (4,8). The first of these involved a transient experiment with a rapid exponential voltage decay output. The second illustration involved application of ensemble averaging to the analysis of an experimental output which provided a repetitive voltage waveform with superimposed large amplitude random noise.

The exponential decay waveform had a time constant the order of 0.1 sec. Data were taken at a constant rate for a fixed length of time, and no processing of the data was carried out. The data were acquired, stored, and the digitized data printed out at the completion of the experiment. The program used is given in Figure 6.

The waveform for the ensemble averaging experiment had a fundamental frequency the order of 10 Hz., with superimposed large amplitude random noise. The experiment was repeatable and could be triggered by computer output. The programming objectives included repetitive initiation of the experiment, synchronization of data acquisition with experimental output, and repetitive coherent summation of the digitized waveforms to accomplish ensemble averaging. Finally, when the experiment was completed the averaged data were normalized and plotted on the teletype terminal. The computer program used is given in Figure 7. Figure 8 shows the original output waveform and the results of ensemble averaging.

Observations.

It can be seen from the above examples that actual implementation of the PRTB system is relatively straightforward. Moreover, for all its simplicity, considerable experimental measurement capability exists in the system.

The requirements for implementing the PRTB system in the undergraduate laboratory include the following:

(1) an introduction to the BASIC language for computational purposes. This requires 1 to 2 hours of lecture time. It also requires that students have access to the laboratory computer (or other computer facilities providing BASIC capability) during the first few weeks of the semester to handle homework and laboratory computations.

(2) an introduction to the fundamental concepts of on-line computer operation. This requires about 6 hours of lecture. These introduce the student to the whole field of on-line computer applications and the technological details with which they must be familiar to implement this approach in the laboratory. These details include an introduction to the fundamentals of digital logic, timing, and synchronization. This involves the discussion of simple gates (AND, OR, NOT), the flip-flop, the one-shot, and the analog switch. The functional characteristics of these devices, as well as of analog-to-digital and digital-to-analog converters and voltage amplifiers, for interface design are defined. No attempt is made to provide a rigorous understanding of the electronic principles or detailed circuitry for these devices. The emphasis is on the student being able to recognize and use the fundamental characteristics of these devices for interface design.

(3) finally, the actual laboratory assignments are carried out. These assignments are designed to introduce the student to the analytical technique and methodology, just as would be done in

the absence of the laboratory computer instrumentation. In addition, the student is expected to utilize the on-line computer as a data acquisition and data processing tool.

The final question to be considered here is what specific benefits are derived by the student from the use of the laboratory computer in on-line experimentation. First of all, the student is obviously exposed to state-of-the art technology in laboratory experimentation. Secondly, the student is encouraged to use more rigorous data processing approaches than previously feasible for laboratory assignments. Thirdly, the student becomes keenly aware of the factors which limit and define experimental accuracy and precision. Fourthly, the student is able to make a first hand comparison of the effectiveness of conventional and computerized laboratory methodology. And, finally, the student's interest in the science of quantitative chemical experimentation is strongly stimulated.

Future Work.

Several different experiments involving on-line computer studies are currently being developed which are appropriate for incorporation into the undergraduate analytical laboratory. These include kinetic methods of analysis, amperometric titrations, coulometric titrations, potentiometric titrations, spectrophotometric analysis, fast-sweep polarographic analysis, and thermal analysis methods, in addition to synthetic and chromatographic experiments already developed (8,9). Some of these experiments will be included in the introductory course and some in an advanced analytical course (senior level at Purdue).

It is also desired to further improve the PRTB software so that the experimental subroutine calls can be replaced by conversational macro-instructions.

ACKNOWLEDGEMENTS

This work supported by a grant from the National Science Foundation Office of Computing Activities, Grant No. GJ-428. The authors wish to thank Dr. David O. Jones for technical advice, and Professors L. B. Rogers, H. L. Pardue, and F. E. Lytle for their respective contributions to the suggestion, encouragement, criticism, and technical support of this program.

TABLE I.

DATA ACQUISITION AND CONTROL SUBROUTINES FOR PRTB

SUBROUTINE	FUNCTION	CALL FORMAT
SB1	Initialize I/O. Set up data storage address and acquisition rate for other subroutines. "X(I)" data variable. "F" = data acquisition frequency.	CALL (1,X(I),F)
SB2	Used to start the CLOCK. The output control bit (ENCODE) can be connected to some external logic preceding the CLOCK ENABLE input. (NOTE: SB2 must be preceded by SB1.)	CALL (2)
SB3	Waits for ADC FLAG, indicating conversion completed. One datum is then taken from the ADC, converted to floating point, and saved as X(I).	CALL (3)
SB4	When called this subroutine waits for switch 0 of the console switch register to be set to a "1" before continuing with the next statement in the BASIC program.	CALL (4)
SB5	Outputs one control bit (ENCODE) and then waits for an event FLAG on the specified I/O channel, "C", before continuing. FLAG and ENCODE are cleared before exit.	CALL (5,C)
SB6	Waits for FLAGS on the ADC channel. When FLAG is set, the next statement in the BASIC program is executed. (Must be preceded by SB1 and SB2 if internal CLOCK is used to generate FLAGS.)	CALL (6)
SB7	Takes in complete block of data before returning to BASIC program. (Max. of 250 pts. Must be preceded by SB1 each time it is called.) "T" sets total No. of data pts. taken. Synchronized with start of experiment through external gating of the ENCODE output. Up to 20 KHz data acquisition rate possible.	CALL (7,T)

SUBROUTINE	FUNCTION	CALL FORMAT
SB8	Causes output bit No. "Z" to be set TRUE on specified I/O channel, "C". Z, C specified in decimal. All other bits will be cleared.	CALL (8,Z,C)
SB9	Causes output of analog voltage through the DAC. "D" = mv. output.	CALL (9,D)
SB10	Clears ADC channel. External Flip-flop driven by ENCODE will be cleared also. This can be used to turn off clock, stop experiment, etc.	CALL (10)
SB11	Causes output of binary voltage pattern, "Z", on specified output channel, "C". Binary pattern and channel specified in decimal.	CALL (11,Z,C)
SB12	Inputs status of 16-bit register on channel "C". The floating point equivalent is saved in memory as "Z".	CALL (12,Z,C)
SB13	Inputs status of a specific bit (No. "Z") on channel "C". Value of "S" is made "1" or "0" accordingly.	CALL (13,Z,S,C)

LIST OF REFERENCES

1. J. W. Frazer, Anal. Chem., 40, 26A (1968).

2. N. R. Kuzel, H. E. Roudebush, C. E. Stevenson, J. Pharmaceut. Sci., 58, 381 (1969).

3. S. P. Perone, J. Chrom. Sci., 7, 714 (1969).

4. S. P. Perone, J. Chem. Ed., 47, 105 (1970).

5. J. G. Kemeny and T. E. Kurtz, "BASIC Programming," John Wiley & Sons, N.Y., 1967.

6. H. V. Malmstadt and C. G. Enke, "Digital Electronics for Scientists," W. A. Benjamin, Inc., N.Y., 1969.

7. "Digital Logic Handbook," Digital Equipment Corporation, Maynard, Mass., 1968.

8. D. O. Jones, Submitted for publication.

9. S. P. Perone and J. F. Eagleston, in Press, 1971, J. Chem. Ed.

LIST OF FIGURES

Figure 1. Schematic Diagram of On-Line Computer Configuration.

Figure 2. Synchronization of Clock Output with Computer and Experimental Events.

A. Direct computer control of Clock Enable input.
B. Clock Enable controlled by AND gate output. Clock enabled when both computer command and experimental start are seen at AND gate inputs. The "RUN" flip-flop output follows the status of the experiment; it is set "true" when the experiment starts and "False" when the experiment ends. Thus, the Clock is disabled when the experiment ends.

Figure 3. Typical PRTB Experimental Data Acquisition Set-up.

Previously undefined symbols include:
I/O Bus - Input/Output hardware of the computer, one channel of which is used for data acquisition system.
16-bit duplex register - A Hewlett-Packard #02116-6195 16-bit duplex interface buffer register card, also providing output ENCODE command and input FLAG signal.

Figure 4. Schematic Diagram for PRTB Programmable Clock.

Components:

Device No.	Manufacturer	Description
F910959	Fairchild	Hex inverter-six Level Converters, High Level to +4 to +20 V.
MC7475P	Motorola	Quad latch--4 bistable latches
SN74151	Texas Instrument	Data Selector/ Multiplexer
F930059	Fairchild	4-bit shift register
F900359	Fairchild	Triple 3-input Nand Gate

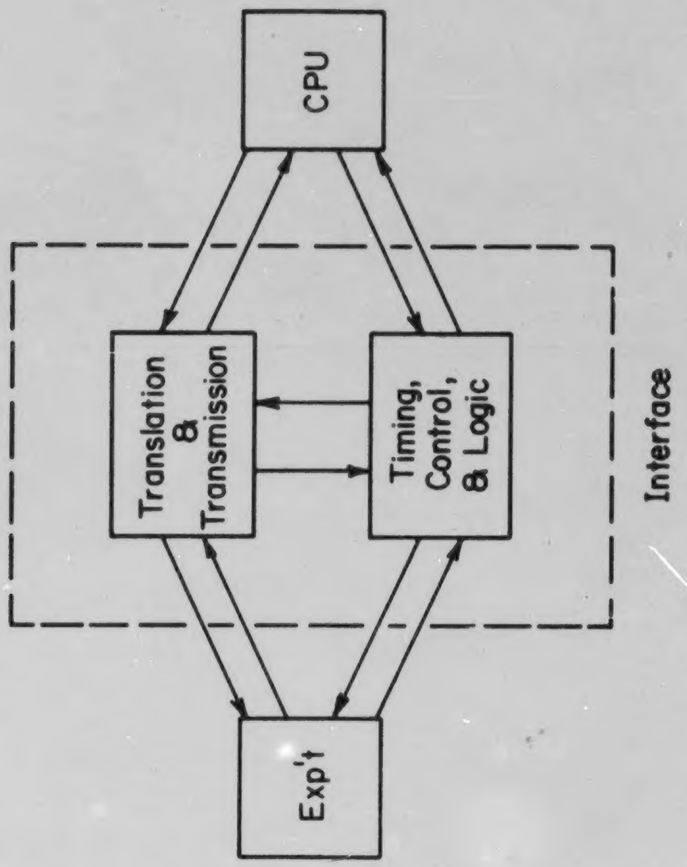
Figure 5. Digital I/O Interface Module for PRTB.

Figure 6. Program for Data Acquisition from Exponential Decay Function.

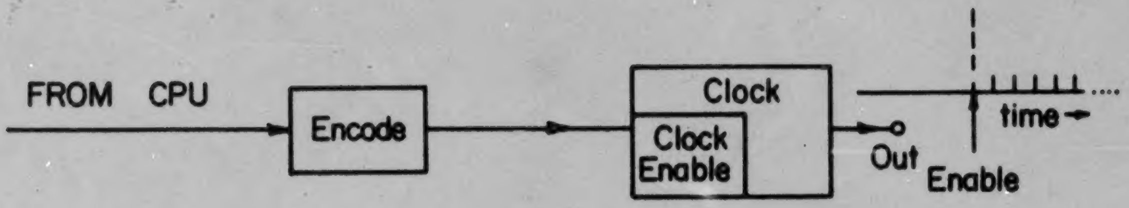
Figure 7. Program for Ensemble Averaging Experiment.

Figure 8. Data from Ensemble Averaging Experiment, plotted on Teletype.

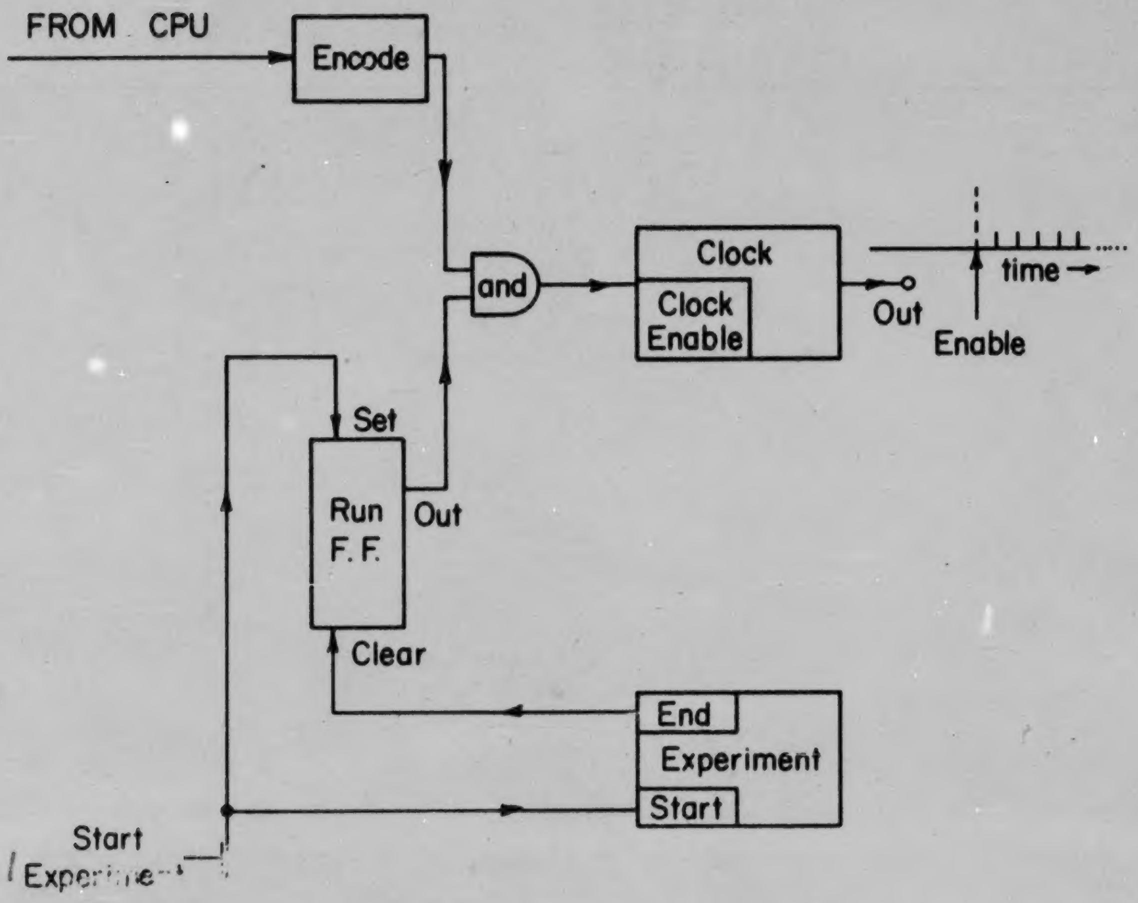
A. Original data (single cycle).
B. Data after 100 averaging cycles.

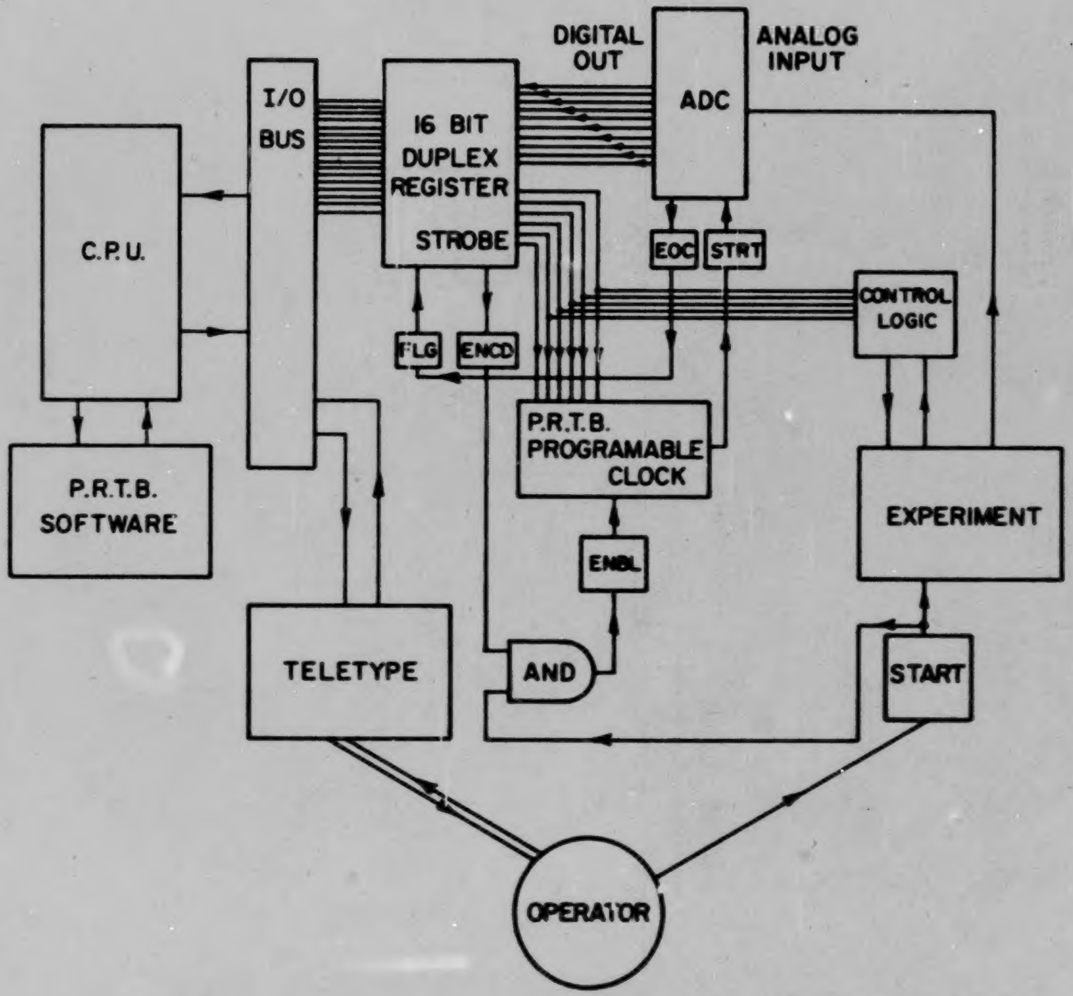

CPU
Translation & Transmission
Timing, Control, & Logic
Interface
Exp't

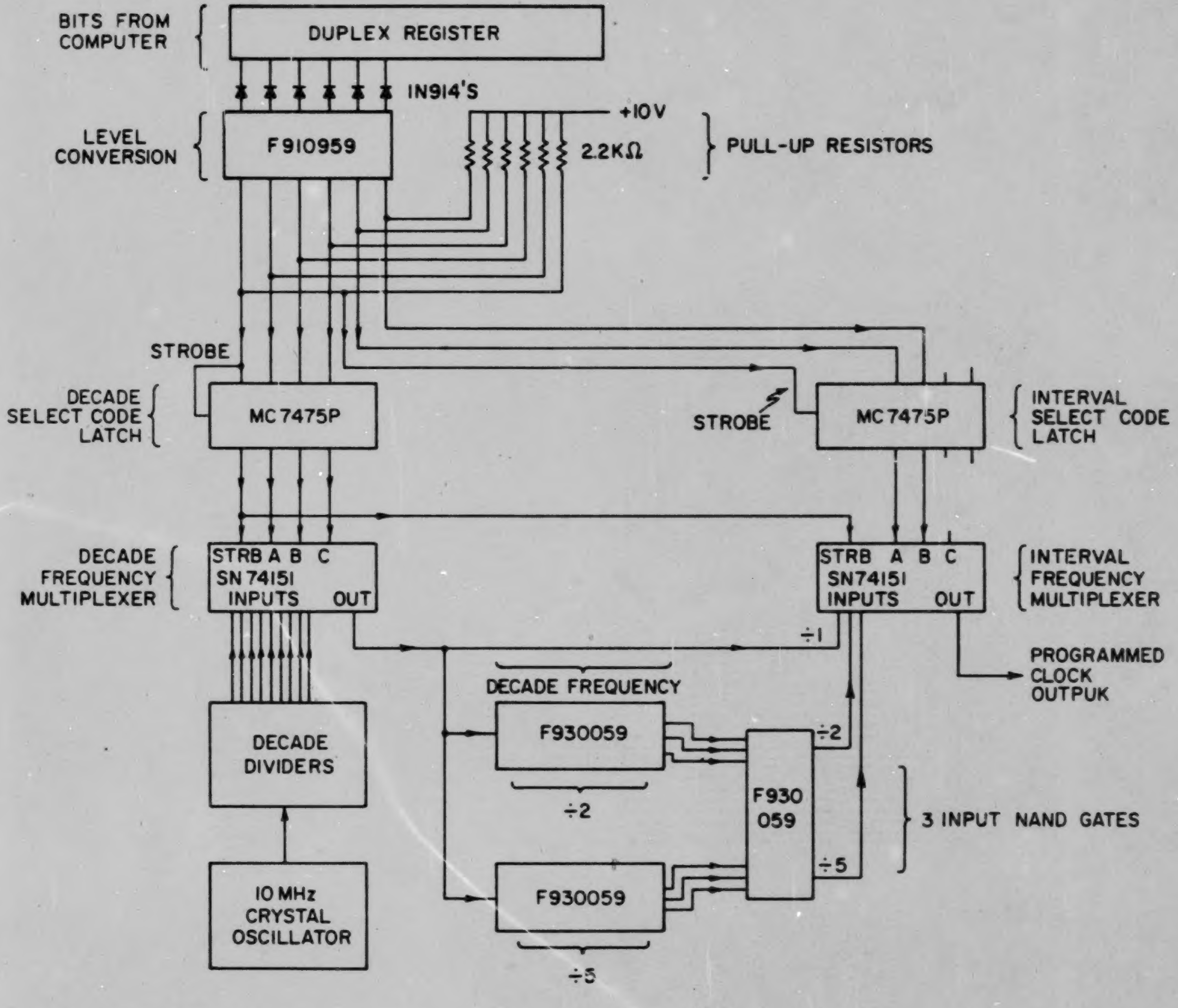
A

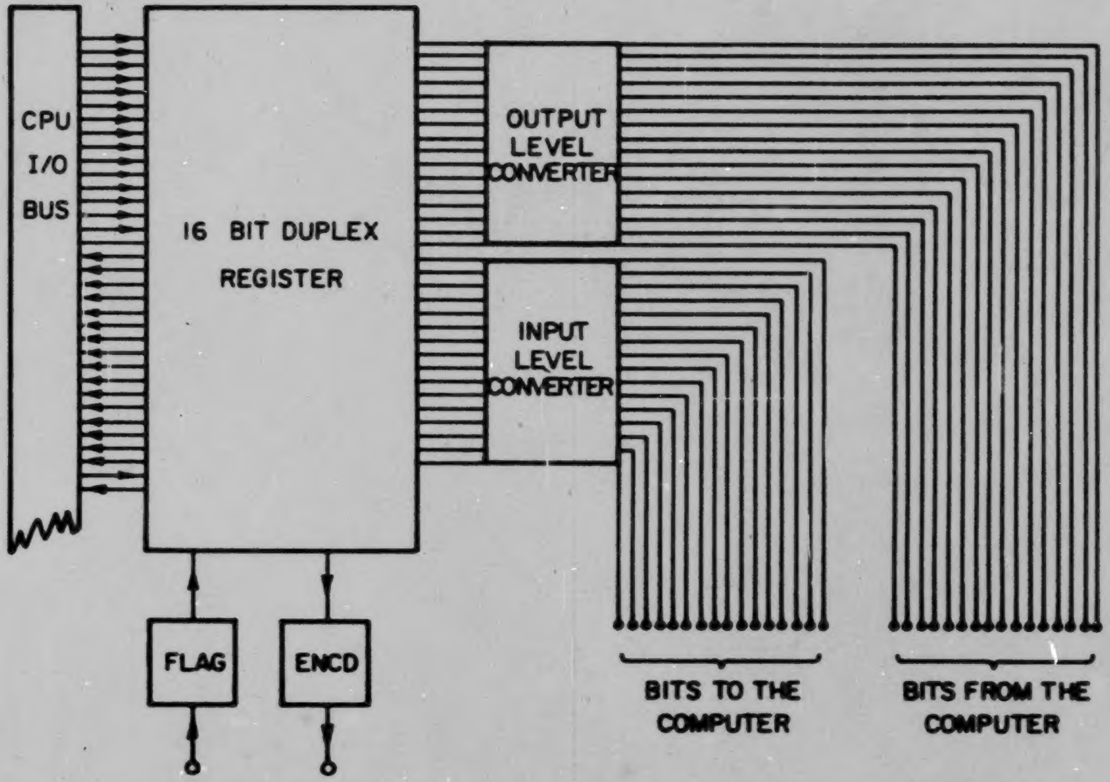

FROM CPU
Encode
Clock
Clock Enable
Out
time
Enable

B


FROM CPU
Encode
and
Clock
Clock Enable
Out
time
Enable
Set
Run F.F.
Out
Clear
End
Experiment
Start
Start Experiment

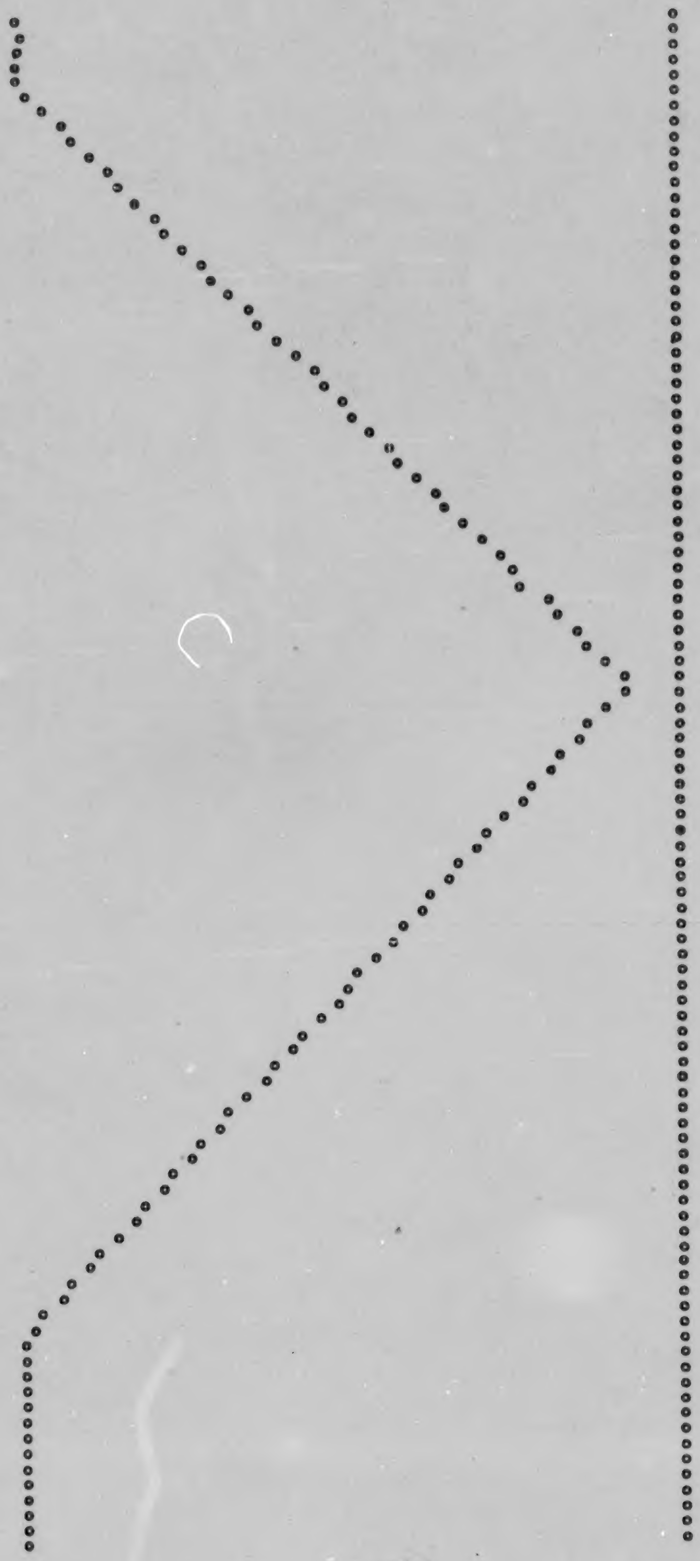

DIGITAL OUT
ANALOG INPUT
I/O BUS
16 BIT DUPLEX REGISTER
STROBE
ADC
EOC
STRT
C.P.U.
CONTROL LOGIC
FLG
ENCD
P.R.T.B. PROGRAMABLE CLOCK
P.R.T.B. SOFTWARE
EXPERIMENT
ENBL
TELETYPE
AND
START
OPERATOR

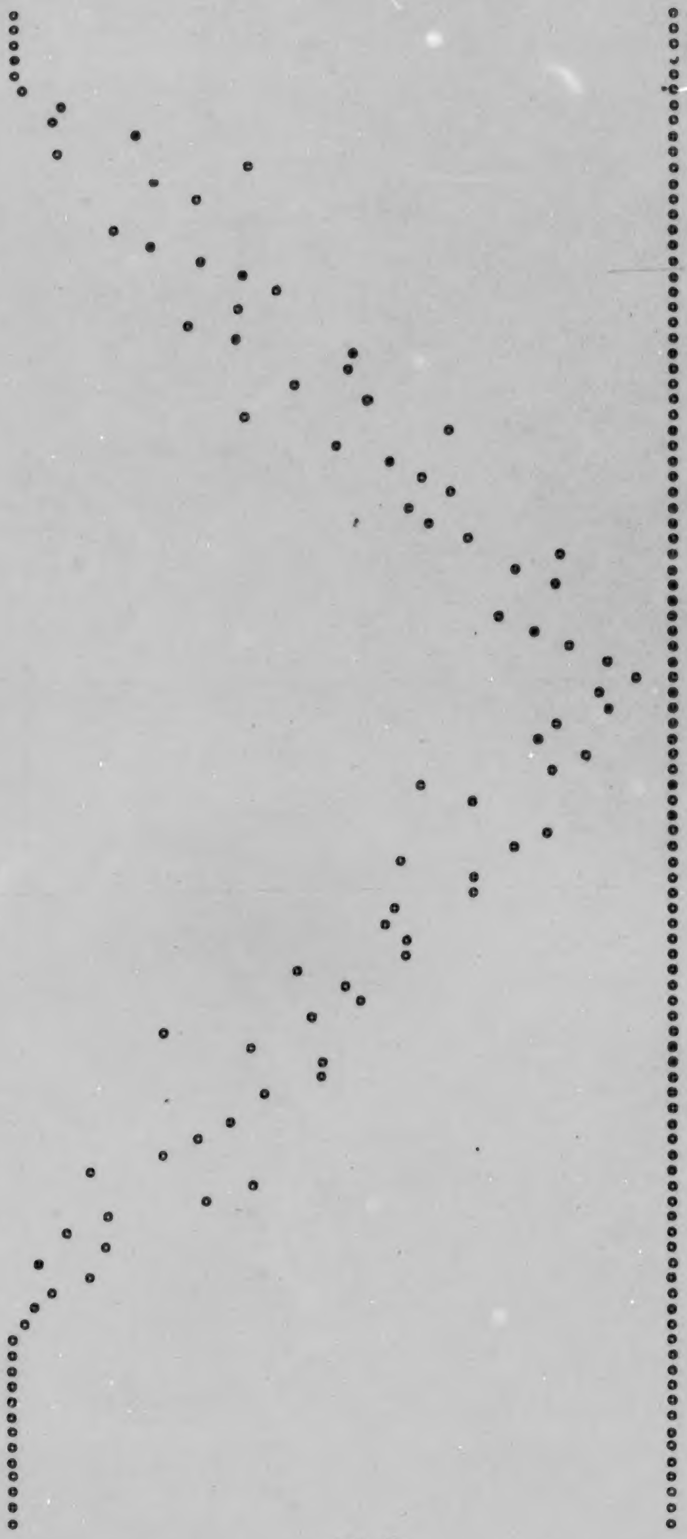

BITS FROM COMPUTER
DUPLEX REGISTER
IN914'S
+10V
LEVEL CONVERSION
F910959
2.2KΩ
PULL-UP RESISTORS
STROBE
DECADE SELECT CODE LATCH
MC7475P
STROBE
MC7475P
INTERVAL SELECT CODE LATCH
DECADE FREQUENCY MULTIPLEXER
STRB A B C
SN74151
INPUTS OUT
STRB A B C
SN74151
INPUTS OUT
INTERVAL FREQUENCY MULTIPLEXER
÷1
PROGRAMMED CLOCK OUTPUK
DECADE FREQUENCY
DECADE DIVIDERS
F930059
÷2
F930 059
÷2
3 INPUT NAND GATES
÷5
10 MHz CRYSTAL OSCILLATOR
F930059
÷5


CPU
I/O
BUS
16 BIT DUPLEX
REGISTER
OUTPUT
LEVEL
CONVERTER
INPUT
LEVEL
CONVERTER
FLAG
ENCD
BITS TO THE
COMPUTER
BITS FROM THE
COMPUTER

```
READY
LIST
10 REM THIS IS A PROGRAM THAT WILL PRINT OUT THE DECIMAL NUMBER
20 REM EQUIVALENT OF AN ANALOG EXPONENTIAL DECAY CURVE.
30 DIM X(250)
40 LET F=100
50 LET T=250
60 CALL (1,X(1),F)
70 CALL (7,T)
80 FOR I=1  TO  250
90 PRINT X(I);
100 NEXT I
110 END
```

```
READY
LIST
1 REM THIS PROGRAM IS AN ENSEMBLE AVERAGING ROUTINE--FOLLOW
2 REM THE DIRECTIONS PRINTED OUT AFTER PRESSING RUN.THE
3 REM FREQUENCY ORIGINALLY SPECIFIED IS 1000 HZ.
10 DIM X(100),Y(100)
20 LET F=1000
30 PRINT "THE # OF PTS EACH RUN =";
31 INPUT T
40 PRINT THE # OF RUNS =";
41 INPUT C
50 MAT Y=ZER
60 FOR I=1 TO C
70 CALL(1,X(1),F)
80 CALL(7,T)
90 FOR J=1 TO T
100 LET Y(J)=Y(J)+X(J)
110 NEXT J
120 NEXT I
200 FOR K=1 TO T
210 LET E=Y(K)/(C*10)
220 IF E>72 THEN 300
230 PRINT "*";TAB(E);"*"
240 NEXT K
250 STOP
300 LET E=71
310 GOTO 230
400 END
```





COMPUTER-BASED TEACHING OF ORGANIC CHEMISTRY

Stanley Smith

Department of Chemistry
University of Illinois
Urbana, Illinois 61801

Increased demands for high quality chemistry courses for students with diverse backgrounds and objectives provide a challenge to the development of innovative and more effective teaching methods.

A tutorial procedure, in which each student has the complete attention of the professor, would meet the educational needs, but the many students involved and the high cost of this approach prohibit extensive use of this method. However, the development of computer-based educational systems makes it possible to approximate closely the one-to-one correspondence between a professor and a student and offers techniques for providing many students with essentially individual attention at acceptable cost levels. One of these systems, PLATO (Programmed Logic for Automatic Teaching Operations), developed at the University of Illinois, has been programmed to present various aspects of undergraduate courses in organic chemistry. Unlike films, television lectures, and, all too often, live lectures, PLATO is not a passive learning aid. The student is directly involved. His decisions and his actions control his progress.

In the present PLATO system, each of twenty student terminals is equipped with a television screen and a keyset, which closely resembles a typewriter keyboard. Information displayed on the screen may consist of words, numbers, formulas, graphs, or diagrams drawn by the computer or images selected by the computer from a set of slides. By the superpositioning of computer plotting and slides, complex structures, formulas, and diagrams can be easily presented to the student. In turn, through the keyset, the student asks and responds to questions, supplies parameters for calculations to be performed by the computer, and suggests reagents for chemical reactions which the computer is to illustrate. Since the computer can accept and respond to chemical terminology, it is unnecessary for students to be acquainted with computer programming techniques.

Verification of a correct response or the correction of an error requires less than a second. If the instructor wishes, the student's progress through the lesson material may be controlled by the number of errors he makes. If the student has difficulty, extra help can be provided automatically. A printed record of each student's work is provided for the instructor by the computer, making it possible for the progress of any individual to be reviewed in detail.

A summary of the lessons currently in use is given in Table 1. These programs combine tutorial techniques, drill, and simulated experiments to develop chemical principles.

Table 1. Summary of Organic Chemistry PLATO Lessons

1. Introduction to organic structures and nomenclature.
2. Structure, nomenclature, and reactions of alkenes.
3. Properties, nomenclature, and reactions of alcohols.
4. Aldehyde and ketone chemistry.
5. Introduction to aliphatic synthesis.
6. Multistep aliphatic synthesis.
7. Introduction to electrophilic aromatic substitution.
8. Multistep electrophilic aromatic substitution reactions.
9. Introduction to nmr spectroscopy, chemical shifts, spin-spin coupling, and interpretation of the proton nmr spectra of organic compounds.
10. Carbohydrate chemistry.
11. Simple and fractional distillation.
12. Qualitative organic analysis. Identification of unknown compounds by combination of spectroscopic and chemical techniques.

Of particular interest is the use of simulated experiments. In textbooks and lectures, the results of important experiments are usually given as facts to be learned. Although it is only rarely possible to allow those in a large class to do these experiments, the computer can be programmed so that each student may conduct simulated experiments.

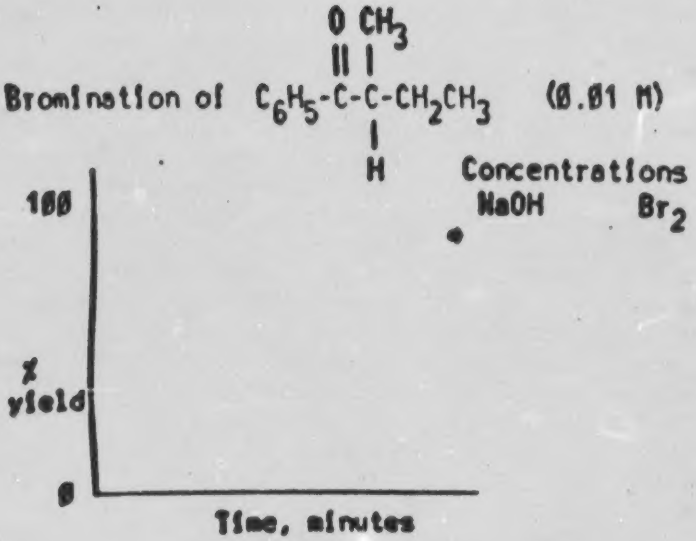
The general concept used in these simulations is to allow the student to ask the computer to perform a series of experiments and report the results. When the student is satisfied that he has devised the correct experiment and understands the results, he may proceed with the program. Of course, he is immediately asked questions which can be answered if the correct experiments have been done. If the student cannot answer the questions, the program indicates the type of experiments he should investigate and directs him to that part of the lesson. It is hoped that this procedure will encourage and reward learning by discovery and analysis, thus providing a depth of understanding which only rarely results from learning by rote.

The bromination of ketones is used here to illustrate this approach to learning. Figures 1-3 are photographs of the television display on a student terminal. Figure 1 shows the screen before any experiments have been run. In previous frames the student has been advised to select values for the concentrations of the two reactants, sodium hydroxide and bromine. In Figure 2, three student-selected experiments have been recorded as graphs of % yield of product vs. time. The first experiment is plotted as a series of ones, the second as twos, etc. It should be noted that experiments 2 and 3 give exactly the same curve. The student may perform as many experiments as he wishes, each experiment being completed in about one second. When the student chooses to go on, the question given in Figure 3 appears. Here he must indicate how each of the reactants affects the rate of formation of the product of the reaction. When an incorrect answer is given, the computer automatically suggests further experiments and returns the student to the display shown in Figure 1. It is important to note that no clues are provided by the format of the questions. The computer has been programmed, through key word judging, to analyze the validity of the student's responses.

Development of the facility to outline suitable schemes for the synthesis of compounds is also an important component of introductory courses in organic chemistry. Even relatively simple molecules can provide a stimulating intellectual challenge to beginning students. However, to encourage imaginative approaches, it is important to allow each individual wide freedom in the choice of starting materials and reactants. In addition, rapid evaluation of the correctness of a proposed synthetic scheme promotes interest and confidence.

Algorithms have been devised which permit the PLATO computer system to judge the correctness of a proposed synthesis by computing the structure of the product of each step in the reaction, displaying the structure on the student's screen, and comparing the structure with that of the desired final product. This approach permits a student to verify quickly the correctness of a proposed synthetic scheme even though it may be quite different from that outlined in the textbook or lectures.

The start of a typical program, at the first semester organic chemistry level, is illustrated in Figure 4. The student is first asked to indicate the starting material of his choice. Either formulas, common names, or IUPAC names may be used. The requirement of three carbons or less in the starting material was imposed

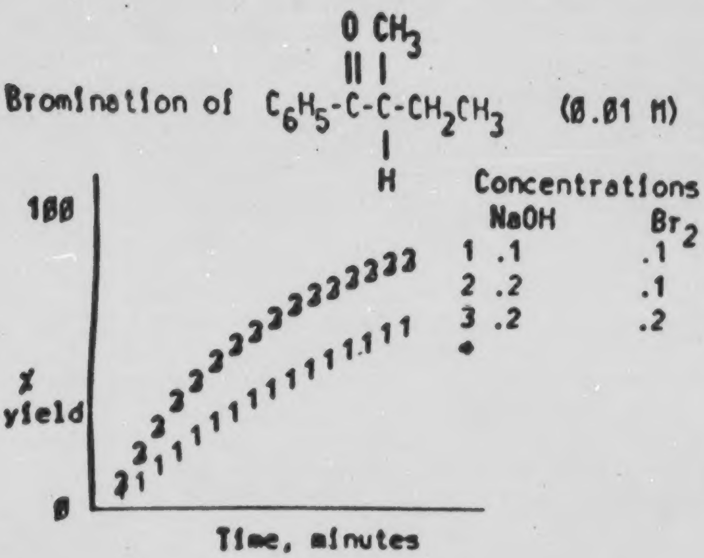


Bromination of $C_6H_5-\overset{\overset{O}{\|}}{C}-\underset{\underset{H}{|}}{\overset{\overset{CH_3}{|}}{C}}-CH_2CH_3$ (0.01 M)

Concentrations

	NaOH	Br_2

Figure 1. Photograph of PLATO screen. To study factors affecting rate of reaction, the student selects various concentrations of NaOH and Br_2.

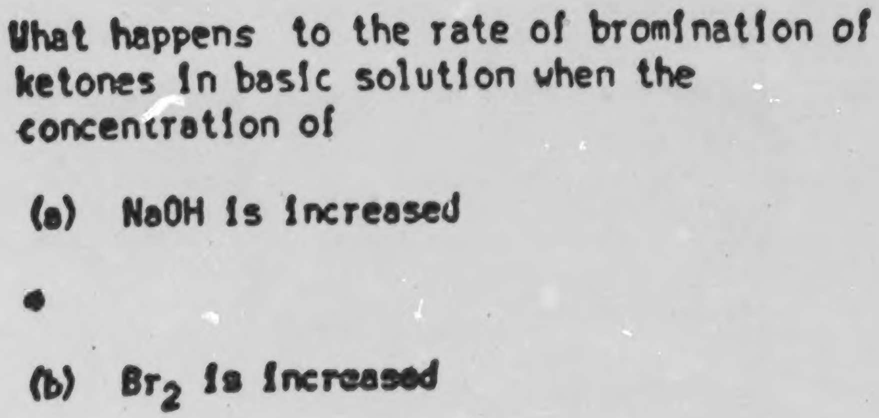


Bromination of $C_6H_5-\overset{\overset{O}{\|}}{C}-\underset{\underset{H}{|}}{\overset{\overset{CH_3}{|}}{C}}-CH_2CH_3$ (0.01 M)

Concentrations

	NaOH	Br_2
1	.1	.1
2	.2	.1
3	.2	.2

Figure 2. The results of three experiments plotted as yield of product vs. time are displayed on the screen. Experiments 2 and 3 give the same graph.



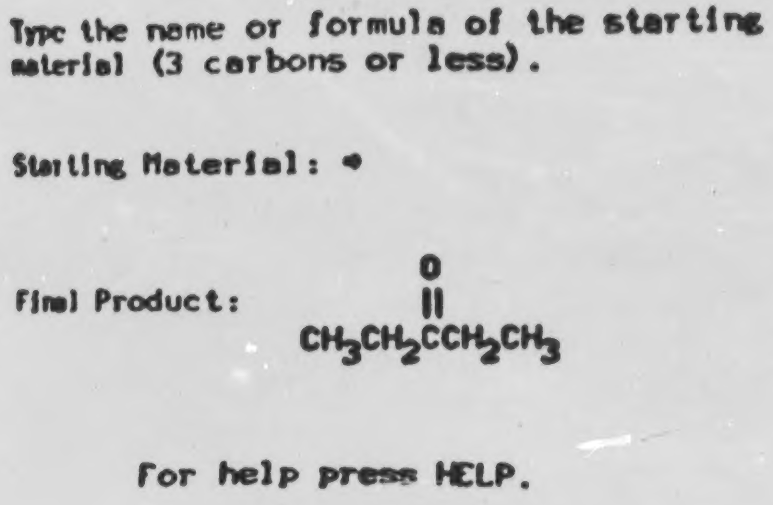
What happens to the rate of bromination of ketones in basic solution when the concentration of

(a) NaOH is increased

●

(b) Br_2 is increased

Figure 3. If the correct experiments have been done, Figures 1 and 2, these questions can be answered. Key word judging is used by the computer to recognize a correct response.



Type the name or formula of the starting material (3 carbons or less).

Starting Material: ●

Final Product: $CH_3CH_2\overset{\overset{O}{\|}}{C}CH_2CH_3$

For help press HELP.

Figure 4. Start of aliphatic synthesis. The student must indicate the starting material of his choice.

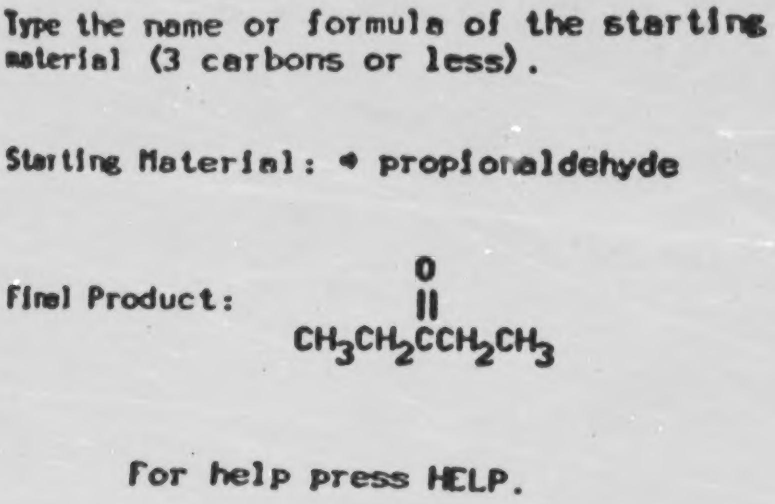
because of the desire, for pedagogical reasons, to restrict these particular problems to compounds with relatively few carbon atoms in order to focus attention on the chemical reactions involved. At present, 53 different starting materials are recognized, each by several names.

The computer responds to a suggested starting material by drawing the structural formula and requesting that the student type the name or formula of the reagents to be used in the first step of the synthesis.

In Figure 5, propionaldehyde has been selected as the starting material and in Figure 6 the reagent for the first step is to be indicated. Any of 134 reagents, which can be either organic or inorganic, may be used in each step. Again, to promote ease of use, each reagent is recognized by several names. Additional synonyms or chemically similar reagents may be incorporated into the lesson in about one minute, even while a class is using the program.

The product of the reaction automatically becomes the starting material for the next step, until the synthesis is complete, Figure 7. Should a given scheme not lead to the product, it is possible to start over at any stage by pressing a single key.

Since there are many acceptable solutions to each problem, upon completion of a synthesis the student is asked whether he wants to work the problem again or try a different problem.



Type the name or formula of the starting material (3 carbons or less).

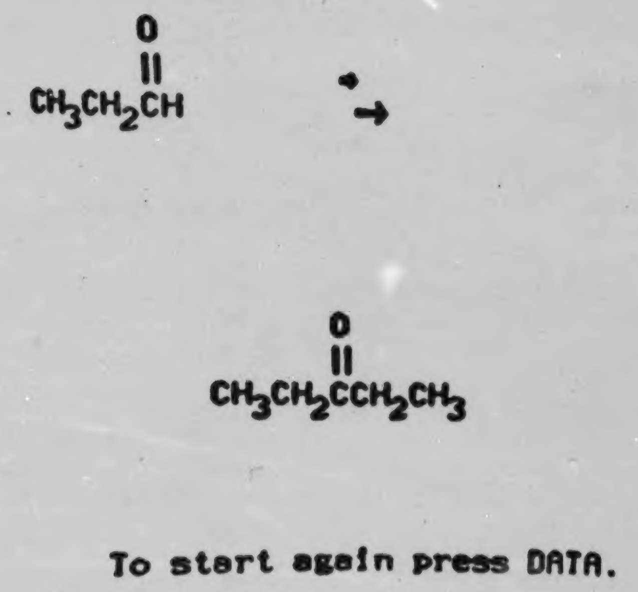
Starting Material: ▸ propionaldehyde

Final Product:

$$\mathrm{CH_3CH_2\overset{\overset{O}{\|}}{C}CH_2CH_3}$$

For help press HELP.

Figure 5. Propionaldehyde has been selected as the starting point for the proposed synthesis of 3-pentanone.

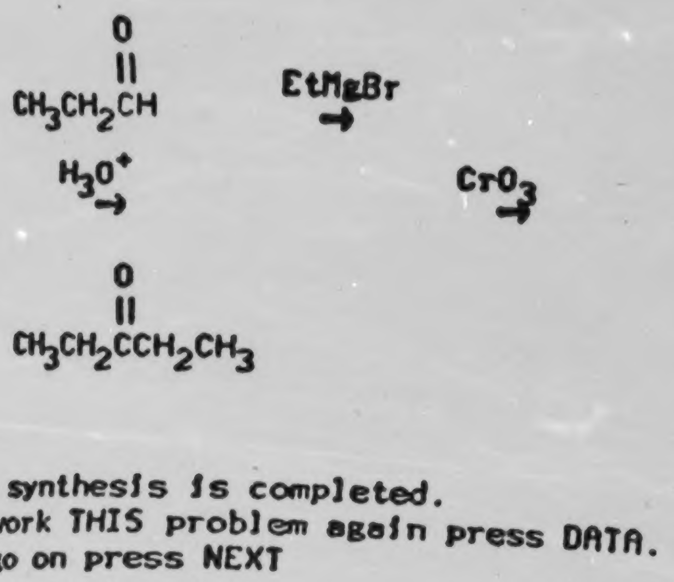


$CH_3CH_2\overset{\overset{O}{\|}}{C}H$ $\rightarrow$

$CH_3CH_2\overset{\overset{O}{\|}}{C}CH_2CH_3$

To start again press DATA.

Figure 6. PLATO has drawn the structural formula of the selected starting material. The reagent for the first step in the synthesis is now to be indicated.



$CH_3CH_2\overset{\overset{O}{\|}}{C}H$ $\xrightarrow{EtMgBr}$

$\xrightarrow{H_3O^+}$ $\xrightarrow{CrO_3}$

$CH_3CH_2\overset{\overset{O}{\|}}{C}CH_2CH_3$

The synthesis is completed.
To work THIS problem again press DATA.
To go on press NEXT

Figure 7. Summary of the completed synthesis of 3-pentanone.

Syntheses involving electrophilic aromatic substitution have been programmed in a similar way, with the computer again showing the product at each stage of the student's synthesis; and, where several isomers are formed, e.g., ortho-meta-para mixture, the composition is computed from appropriate linear free energy relationships and displayed on the screen so that the student can develop a general awareness of isomer distribution in these reactions.

The current introduction of the PLATO IV system, which utilizes plasma panels, will make it possible for up to 4,000 terminals to be operated from a central computer.

Acknowledgement. This work was made possible by the cooperation and assistance of Professor D. L. Bitzer, Director of the Computer-Based Educational Research Laboratory, his staff, Mr. Paul Tenczar, who developed many aspects of the programming procedures used, and Professor D. Y. Curtin.

The Computer-Based Educational Research Laboratory is supported in part by the National Science Foundation under grants GJ81 and GJ974 and the Advanced Research Projects Agency under grant ONR-NONR 3985(08).

Computer-Assisted Instruction in Chemistry at Pitt

E. E. Tratras and K. J. Johnson
Department of Chemistry
University of Pittsburgh
Pittsburgh, Penna. 15213

Sixteen tutorial-drill CAI programs were made available to freshman chemistry students during the fall term of 1970. They have been implemented on an IBM 360/50 computer, using the Pitt Time-Sharing System (PTSS). PTSS supports the CAI language CATALYST/PIL. CATALYST (Computer Assisted Teaching and Learning System) is a lesson-designer oriented language developed at Pitt. PIL (Pitt's Interpretive Language) is a JOSS - derived conversational language. The CATALYST/PIL interface provides a nearly ideal CAI language for both lesson designers and students.

Arithmetic Features

One of the most attractive features is the ease with which students can make chemical calculations. In addition to the standard verbal and numeric answers, CATALYST/PIL will accept arithmetic expressions. For example in an ideal gas law problem the answer might be typed as follows:

```
>785*2.5*(50 + 273)/(273*760*.75)
```

The answer to an equilibrium hydrogen ion concentration problem would be given (approximately) by

```
>(0.185*1.8e-5)**(.5)
```

If the student is interested in the magnitude of the answer or if the problem involves more than one arithmetic expression, the PIL processor is made available when the student types "calc". The following is part of a lesson drilling percent composition:

The computer dialog is in upper case.

```
                                  .
                                  .
                                  .

WHICH OF THE FOLLOWING MINERALS IS RICHER IN COPPER?
          MINERAL                  MOLECULAR WEIGHT
          (Cu)5Fe(S)4              502 GRAMS/MOLE
          (Cu)2S                   159 GRAMS/MOLE
(HINT:  DETERMINE THE PERCENT  Cu IN BOTH MATERIALS)
```

```
>calc
READY
>type 5*63.5*100/502
5*63.5*100/502 = 63.24701
>type 2*63.5*100/159
2*63.5*100/159 = 79.87421
>return
RESPOND TO LAST QUESTION
>Cu2S
CORRECT.  NOW, . . .
```

The response "calc" brings PIL to the foreground. PIL types "READY:", and the student can use the desk calculator mode as is done here, load a prewritten program or write and execute a PIL program. When he has the answer he reactivates CATALYST by typing "return". CATALYST then asks for his response to the last question and continues.

PIL arithmetic functions are very close to English. For example,

.
.
.

```
WHAT IS THE (APPROXIMATE) pH OF A 0.147 MOLAR ACETIC ACID SOLUTION?
(Ka = 1.8E-5)
>calc
READY:
>type - the log of the square root of (0.147*1.8e-5)
- the log of the square root of (0.147*1.8e-5) = 2.788705
>return
RESPOND TO LAST QUESTION
>2.79
VERY GOOD.  NOW, . . .
```

Another useful arithmetic feature of PIL for CAI applications is the pseudo random number generator. At any point the lesson designer can acquire a random number uniformly distributed between zero and one. To distribute the number between arbitrary limits a and b, the simple algorithm

$$r_{ab} = a + (b-a)\ r_{01}$$

is used. For example if a random temperature between 273° K and 373° K were desired, the expression

$$T = 273 + 100 * r_{01}$$

would be coded in the CAI lesson. Then if the random number generator returned a value of zero, T would be assigned the value 273. If r_{01} is one, T becomes 373.

We have utilized these arithmetic features of the CATALYST/PIL interface in designing the CAI molecules. Most of the lessons drill numerical problems in the areas of stoichiometry and aqueous equilibria. These problems frequently include PIL-generated random numbers. Thus no two students will receive the same numerical problem. The format of the problems are fixed, but the values of the parameters (e.g. temperature, mass, concentration, etc.) vary randomly.

The sixteen CAI lessons are described briefly below.

Name	Description
Stoich: 1	Seven questions reviewing molecular weights, gram-mole and molecule-mole conversions, empirical formulas and percent composition problems.
Stoich: 2	Four problems reviewing molecular weights, gram-mole, molecule-mole, and atom-mole conversions.
Stoich: 3	Four problems dealing with percent composition of metal chlorides, molecular weight of an element from its isotopic mass and abundance, and the empirical formula of a compound.
Stoich: 4	Five problems on the relationship between moles and gram-atoms, use of Avogadro's number, and weight-weight stoichiometry.
Expert: 1	Simulation of stoichiometry experiment to find the empirical formula of a copper and sulfur compound; consisting of two parts, an experimental section and a section of questions relating to the properties of the substances used and experimental techniques.
Gas: 1	Six problems on pressure, volume and temperature relationships of ideal gases.
Gas: 2	Four problems covering gas laws, Graham's law of diffusion and Dalton's law of partial pressures.

Collig: 1	Three problems on colligative properties of solutions: freezing point depression, boiling point elevation, molal solutions and molecular weight determination.
Bonding: 1	Twenty-one questions reviewing periodic properties of the elements, quantum numbers, magnetic properties; no calculations (periodic table is necessary).
Faraday: 1	Ten problems on the application of Faraday's law: mole-faraday, amp-coulomb, current-time conversions and the amount of material involved in oxidation-reduction reactions.
Aqeq: 1	Seven questions discussing acids, properties of electrolytes, dissociation and law of mass action; no calculations.
Aqeq: 2	Four problems covering hydrolysis reactions, dissociation constants, mass action expressions.
Aqeq: 3	Four problems calculating hydrogen ion concentration, pH, hydroxide ion concentration, pOH, and the dissociation constant of an acid.
Aqeq: 5	Two questions and two problems discussing properties of buffer solutions, concentrations of species in solution, and diprotic acids.
Aqeq: 6	Three problems on solubility calculations.

Many freshmen have difficulty with chemistry because they are unwilling to master the required chemical arithmetic. This CAI facility eliminates the drudgery of working problems because the students use the computer to do the calculations. We believe that this teaching aid will make learning chemistry more interesting for average and below average students.

Other Features

These lessons are written so that at any question, if the student needs more information, he may ask for "help". The following example from Aqeq: 6 illustrates this feature along with the random number and arithmetic features.

⋮

```
EXERCISE:  THE IONIC SOLID MX2 DISSOCIATES AS FOLLOWS:

                   MX2<====> M(++) + 2X(-)
```

```
IF THE SOLUBILITY PRODUCT, KSP, IS 6.409E-16,
CALCULATE THE SOLUBILITY OF MX2 IN PURE WATER.

>help
WANT SOME HELP AT THIS POINT?

LET Z EQUAL THE SOLUBILITY OF MX2 IN MOLES/LITER.
THEN IT FOLLOWS THAT IN THE SATURATED SOLUTION,

        (M(++)) = Z MOLES/LITER
        (X(-)) = 2*Z MOLES/LITER

AND (M(++))*(X(-))**2 = KSP

NOW TRY THE PROBLEM AGAIN.

>(6.409E-16/4)**(1/3)

RIGHT.  NOW, . . .
```

Here the value 6.409E-16 (6.409×10^{-16}) is a random number uniformly distributed between 4.670E-16 and 4.670E-10. The student asks for help, receives a hint and is invited to try again. He represents the answer by the arithmetic expression, (6.409E-16/4)**(1/3), which is equivalent to $[\frac{6.409\times10^{-16}}{4}]^{1/3}$.

The student is given two or three chances to respond correctly. If he fails to get the right answer the computer supplies it. Also, hints are sometimes automatically given as in the next example.

.

.

.

```
IF YOU REACT 1,254E+01 GRAMS OF SILVER AND 6.274E+00 GRAMS OF SULFUR
WHAT WEIGHT OF SILVER SULFIDE CAN BE FORMED?

  (MOLECULAR WEIGHT OF (ag)2S=247.9 G/MOLE)

>help

     2 Ag + S ----> Ag2S

CALCULATE THE G-ATOMS OF SILVER AND SULFUR.  DETERMINE WHICH REAGENT
IS LIMITING AND WHICH IS IN EXCESS.

>0

CHECK TO SEE IF YOU CALCULATED THE G-ATOMS OF EACH ELEMENT
CORRECTLY:

     1.254E+01 G * 1 G-ATOM/107.87 G = 1.163E-01 G-ATOMS Ag
     6.274E+00 G * 1 G-ATOM/32.1 G = 1.960E-01 G-ATOMS S

THEREFORE SILVER IS THE LIMITING REAGENT.
```

```
>0

THE SOLUTION IS:

WT. OF (Ag)2S+WT. OF Ag IN GRAMS * MW OF (Ag)2S)  /  (2*AW OF Ag)

    = 1.254E+01 G * 247.9 G/MOLE * 1 MOLE/(2*107.87 G)
THE ANSWER IS 1.440E+01 GRAMS.

YOU HAVE MADE  6 MISTAKES.
 YOU HAVE NOT COMPLETED THE ENTIRE LESSON.
 PLEASE REVIEW SOME MORE AND COME SEE ME AGAIN SOON!

HAVE A GOOD DAY!
M:END OF JOB
```

Other features including "skip", "back" and "comment", are being coded into the CAI lessons for more versatility. If a student wishes to skip the question just presented, he can type "skip" or "+1" and he is branched to the next question. The back feature allows a student to go back question by question if he wishes to review a previous problem. He accomplishes this by typing "back" or "-1". The "comment" feature enables a student to type his opinion of the lesson, or any suggestions at the terminal while thoughts are still fresh in his mind.

Also, a complete management system is being developed to provide feedback on lesson use. Special PIL datasets will be loaded when execution of a CATALYST lesson begins. These PIL datasets will record: which questions are answered correctly or incorrectly; how often a help or hint is used, and how effective it is (i.e., whether the question was answered correctly or incorrectly after the hint is presented); normal and abnormal termination of a lesson; the total number of right and wrong responses; the number of times each question was skipped or repeated; and the comments and suggestions typed by the student. This information will pinpoint problem areas in the lessons and help us create more effective teaching modules.

Evaluation

So far student feedback has been encouraging. The students enjoyed using the lessons and are asking for more programs. Student opinion obtained from an attitude questionnaire indicates that the time required to use the programs was well spent and that CAI is an effective teaching tool. Students also recommended continued development of the program in freshmen chemistry and that they would participate in similar programs in other courses if given the opportunity. The following student comments are taken from the CAI questionnaire:

"I found working with the computer very interesting and enjoyable. Because of this experience..., I am thinking of taking some courses in computer programming."

"The idea of computer-assisted [instruction] programs is great..."

"Once you've read the material, and understand it (outside of the computer room), I feel it can be an invaluable study aid... Thank you for the opportunity."

"It gives me more confidence in my work and motivates one more. I find that I'm enjoying chemistry a lot more because of it."

"I sincerely hope that the computer program will be continued and expanded for the rest of Chem. 11 and onto Chem. 12 material."

"...I am really glad that...CATALYST was offered to me and I hope that more CATALYST will be offered for the rest of Chem. 11 and Chem. 12."

"I have also found this method very effective in my biology course."

"The computer is very witty!"

Predictably, the students most enthusiastic about CAI are the A and B students. Yet the emphasis is on tutorial-drill lessons for average and below average students. Most of the students in this latter group "can never find the time" to locate a terminal and try out the programs. It is hoped that with increased availability of terminals and with the development of a more complete and effective library of CAI programs, this motivational problem will be solved. We plan to continue the CAI project at Pitt. We think that with care we can provide our freshman chemistry students with a learning aid that will make the study of chemistry more interesting and exciting. And we think we can provide the instructer with a tool that will make his teaching more effective.

Direct Classroom Use of a Time-Sharing Computer in Chemical Education

Philip E. Stevenson
Department of Chemistry
Worcester Polytechnic Institute

I. General Considerations

There are many topics in the undergraduate chemistry curriculum where complicated mathematical formulas are necessary for quantitative descriptions of phenomena. These topics range from ionic equilibrium in freshman chemistry to advanced areas in quantum mechanics and statistical mechanics. While teaching these topics, the instructor equipped only with the conventional blackboard and chalk simply cannot demonstrate quantitative application of these formulas, except perhaps by reference to previously calculated quantities, or by the use of hand-waving approximations (as in ionic equilibria), or by the use of very simple idealized cases (such as the Ideal Gas Law).

However, with the use of special computer programs, previously written, debugged, and loaded into a time sharing computer system (such as the RCA Spectra 70/46 at WPI), an instructor with a remote time sharing terminal at his side can demonstrate these quantitative calculations to his classes. He need only type in a line or two of input and wait a moment for the results. The student question, "What happens if we change the value of ... to ...?" is easily answered.

In this paper I will discuss a number of such programs that I have written, some with the assistance of WPI undergraduates. Completed programs include "DIATH2," for computation of the binding energies of H_2 and H_2^+, "SHMO" which performs Hückel MO calculations, "ACID" and "BASE" for simulation of weak acid-strong base, and weak base-strong acid titrations, respectively, "GASLAW" for calculation of pressures by van der Waals' law, and "FEF" for calculation of free energies of reaction for arbitrary temperatures. Programs in development or contemplated include "SALT" for simulation of precipitation titrations and programs in the areas of reaction kinetics and statistical mechanics. First, however, I would like to discuss some of the general aspects of writing such programs.

To begin, one must choose a computer language. At WPI, this choice is among BASIC, FORTRAN IV, and TFOR (time-sharing fortran). Each language has its own advantages and disadvantages. FORTRAN IV is computationally the most powerful of the three. This computation power was necessary for "DIATH2" which required double precision arithmetic for integral evaluation and for "SHMO" which utilizes "EIGEN" from the IBM Scientific Subroutine Package for matrix diagonalization. "ACID" and "BASE" are in FORTRAN IV because they were developed from a previous batch program written in WATFOR. In addition, source programs in FORTRAN IV are normally in card form and can be safely stored and easily be edited, duplicated and distributed. FORTRAN IV has three serious disadvantages, however. Any change in a program requires a batch run, so one must wait for whatever turn around time one's computer center imposes, in order to check the results of the changes. Input format requirements are fairly rigid. For instance, one must right-justify all exponential and integer input. Finally FORTRAN IV load modules in the RCA system are large and non-sharable and sometimes generate paging problems. "DIATH2" for instance has a load module that is close to 100K bytes long. Four users executing "DIATH2" simultaneously would be utilizing about 40% of the available drum pages, and these pages would be swapping in and out of core often enough to decrease significantly the CPU efficiency. TFOR and BASIC on the other hand are not as computationally powerful as FORTRAN IV, but are interactive, so that programming errors can be immediately detected and remedied, have relatively few input restrictions, and use much less core and drum space for object coding.

Having chosen a language, one must next obtain a properly executing algorithm. Often one already has this in a batch program or a program obtained from elsewhere. Then one must devise an input-output scheme that is suitable for interactive execution of the program and suitable for the kind of terminal which one will be using. My personal preference is for teletype because it is the cheapest kind of terminal and because it produces hard copy. Its disadvantages are noisy operation and slow output (10 characters per second). In any event, one wishes to program for minimum input, and relatively small amounts of output. One should keep in mind that the interactive nature of such programs allows one to use output from an earlier part of the program to influence input to a later part of the program. If one has written his program in FORTRAN IV, one must provide some way of aligning input to the expected input format. I do this by outputting strings of characters just ahead of input lines which show me where to place the input. Finally, it is a good idea to have the program type out instructions for its own use.

The kind of problem one would choose for this sort of interactive program ideally would use a very small number of parameters as input, do lots of computation, and generate a modest number of significant results. I feel that the specific programs which I will now discuss more or less meet these criteria.

II. "DIATH2"-A Program for Demonstrating Chemical Bonding in Hydrogen

H_2 and H_2^+ are traditionally the first molecules encountered by the student of quantum chemistry. At some point in his introduction to the subject, he sees his instructor dutifully derive formulas for the energies of the valence bond function for H_2 (1) and the molecular orbital functions of H_2^+ (2) and H_2^1 (3). He may even encounter the Weinbaum (4) and configuration interaction treatments. However, when it comes to numerical results, all he encounters are tables and perhaps a dissociation curve or two. He never sees nor participates in the process of calculating these results. If for this lack of participation he concludes that quantum mechanics is a "black art", he is not without some justification.

However, with the aid of a computer and a program, "DIATH2", described here, it is possible for the quantum chemistry teacher to add numerical results to his classroom discussion of H_2 and H_2^+. Using a remote terminal connected to a time-sharing computer, he may enter arbitrary values of the internuclear distance, R, and 1s orbital exponent, ζ, and obtain the resulting energies within seconds. He could enter the "book" values of R and ζ, or with somewhat more expenditure of time, search for their optimum values. He could assign the construction of optimized dissociation curves as homework.

The program treats the various wavefunctions for H_2 and H_2^+ which can be constructed from 1s orbitals centered on each nucleus, namely the valence bond function of H_2, the molecular orbital functions for H_2^+ and H_2 and the simplest configuration interaction function for H_2. The Weinbaum treatment is omitted, since it is formally identical to the configuration interaction treatment. The program, written entirely in FORTRAN IV, takes the input values of R and ζ, converting R into atomic units, if necessary, and evaluates all the necessary integrals involving components of the Hamiltonian operators over 1s orbitals (Eq. (1)) on the two nuclei.

$$1s = \zeta^{3/2}\pi^{-1/2}e^{-\zeta\rho} \tag{1}$$

The program then evaluates the electronic, total, and binding energies for the valence bond wave function, the molecular orbital wave functions for H_2^+ and H_2, and the configuration interaction wave function[2]. At the user's option, any or all of the output may be typed at the terminal for immediate inspection. Any output not typed is printed off line for later use. An auxiliary subroutine "PARAB", is available which constructs a parabolic curve when given three points. It also gives the extreme point of the curve. This is useful when one is optimizing for ζ or R and desires a quadratic interpolation. The use of the input distance R = 0.0 results in output of the electronic energies of He^+ and He.

I have used this program in my Molecular Orbital Theory course. Student enthusiasm was high, and number of points were brought out and clarified which are often confusing or not appreciated. These points included the incorrect dissociation of the molecular orbital function, the united atom picture for electronic energy, and the role of orbital contraction. Application of the variation theorem was rather effectively demonstrated. However, in the course of an eighty minute class period, there was only time enough for a thorough discussion of H_2^+.[3] Thus in future years, I plan to assign homework problems involving student use of the program, as well as use it for demonstration purposes.

III. Ionic Equilibria Programs

Ionic equilibrium is traditionally one of the most difficult topics in the freshman chemistry syllabus. The combination of complex relationships among the variables, the need for such complicated mathematical techniques as the method of successive approximations, and the need to recognize which variables can be neglected in which equations proves to be the downfall of many a student. Demonstration calculations, though pedagogically necessary, are tedious and time consuming for an instructor to perform in class.

We are writing a series of demonstration programs which simulate titrations. "ACID" computes a pH vs. volume of base added curve for the titration of a monobasic, dibasic, or tribasic weak acid with a strong base. "BASE" is the analogous program for a monoacidic, diacidic, or triacidic weak base. Both "ACID" and "BASE" can also give concentrations of all substances at any point in the titration. "SALT", which is still in development, will simulate precipitation titrations in which the kinds of ions produce slightly soluble 1:1, 2:1, 3:1, or 3:2 precipitates.

One might better appreciate the value of a simulated titration if one recalls that titrations unify the problems of weak acid and base pH's, buffers, hydrolysis, salt solubilities, and common ion effects. A detailed discussion of "ACID" will help clarify this. One deals with the equilibrium of the weak acid HA with its ions;[3]

$$HA(aq) = H^+ + A^- \qquad (2)$$

$$\frac{[H^+][A^-]}{[HA]} = K_a \qquad (3)$$

the self dissociation of water;

$$H_2O = H^+ + OH^- \qquad (4)$$

$$[H^+][OH^-] = K_w = 1.0 \times 10^{-14} \qquad (5)$$

the conservation of the initial amount of acid;

$$[HA] + [A^-] = F_a \left(\frac{V_a}{V_a + V_b} \right) \qquad (6)$$

and the conservation of electric charge;

$$[H^+] + [Na^+] = [OH^-] + [A^-] \qquad (7)$$

The Na^+ ion concentration is known to be:

$$[Na^+] = F_b \left(\frac{V_b}{V_a + V_b} \right) \qquad (8)$$

Eqs. (6),(7), and (8) assume that a given point in the titration is attained by mixing V_a ml of weak acid of formality F_a with V_b ml of NaOH of formality F_b. The program, in fact, assumes $V_a = 50.0$ ml and accepts F_a, F_b and K_a as input parameters. V_b is the independent variable in the titration and is automatically incremented from 0.0 ml to a point 20% past the equivalence point. The algebraic solution of Eqs. (3), (5), (6), (7), and (8) is programmed. The variables, $[OH^-]$ and $[A^-]$ are thus eliminated from Eq. (7) and the resulting cubic equation in (H^+) is solved by the Newton-Raphson method.

One would, of course, never go through this complete solution in a hand calculation, but would instead neglect certain quantities in Eqs. (6) and (7) and then plug the equations into Eq. (3), resorting to the method of successive approximations as a last resort. However, most freshmen lack the chemical intuition to be able to know what quantities to neglect and when to neglect them. Since "ACID" calculates the concentrations of all the ions in solution without neglecting any of them, a freshman chemistry instructor can use it in class to give his students some feeling for the numbers resulting from ionic equilibrium calculations. Thus students could begin to acquire the chemical intuition which they need in order to master the subject of ionic equilibrium.

I have used "ACID" experimentally in my two recitation sections in CH102 (freshman chemistry) at WPI. An ASR-33 teletype was set up at the side of a 30 seat classroom using a TV camera and monitor to show the results to the class. Communication with the RCA Spectra 70/46 computer at the Worcester Area College Computation Center was by telephone. I spent approximately 10 minutes at the beginning of class discussing Eqs. (2) through (8). Then I obtained the titration curve of 50.0 ml of 0.1F acetic acid with 0.1F NaOH and drew attention to the equivalence point pH of 8.72. The specific problems of the weak acid solution, the 1:1 buffer and the hydrolysis of the salt of a weak acid problem were then demonstrated by requesting computation of the concentrations of all species in solution at 0.0 ml, 25.0 ml, and 50.0 ml respectively in the titration. For these problems, all the standard approximations are valid and the program convinced the students that the concentrations of neglected species are indeed at least several orders of magnitude smaller than those of the major species. There remained time for only one additional titration. In one class, titration of a moderately strong weak acid ($K_a = 5 \times 10^{-3}$) was demonstrated; in the other the titration of a very very weak acid ($K_a = 1.0 \times 10^{-12}$) was demonstrated. In these two cases at least one approximation breaks down in each of the standard problems. For example, in the hydrolysis of the salt of the very very weak acid, the self-ionization of water (Eqs. (3) and (4)) is significant. These breakdowns were clearly demonstrated.

Student response to the program is difficult to gauge at the time of this writing (the third day following the demonstration). The students asked very few questions and there was little spontaneous response in class. After class, however, one student asked me how to execute the program from the student teletypes; another complained of poor legibility of the numbers on the screen (a minor change in format will alleviate that problem); and a third said the demonstration was fascinating. The real test will be whether the demonstration enables the students to show an improved performance on their examinations. My feeling is that this will happen.

IV. Other Demonstration Programs

a. Quantum Chemistry

I have written an interactive program, "SHMO," which performs simple Hückel Molecular Orbital calculations. Input consists of the matrix dimension, number of π electrons and the Hamiltonian matrix elements in units of β. Teletype output consists of eigenvalues, charge densities, and bond orders for bonded atom pairs. In addition, a complete output list is generated off line. The program makes use of the SSP matrix diagonalization subroutine "EIGEN." In addition, my research level program, "NEMO" (for Non Empirical Molecular Orbitals (5)) is set up for interactive teletype execution. Here too, a selection of output is typed on teletype, and the complete output list is generated off line. I make use of these two programs and "DIATH2" in my quantum theory course.

b. Statistical Mechanics

I have recently assigned a freshman honors lab student to program the following problem: Given a fixed number of identical indistinguishable particles possessing a fixed total energy, and free to move in a spectrum of evenly spaced one particle energy levels; compute the probability distribution of particles in these levels. This program is intended to generate a Boltzmann distribution by simple counting of particles and will compute distributions for large values of N and E following a hand calculated demonstration for small numbers such as N = 3 particles and E = 6 units of energy. The energy spectrum is assumed to consist of evenly spaced levels with separations of one unit starting at $\epsilon = 0$.

c. Thermodynamics

One of my students has just completed a program, "GASLAW," which computes the Ideal and van der Waals' pressures of gases, given

T, V, and N, as input. A table of parameters a and b for common substances is stored in the program, but the user can supply his own values if he wishes. This program can be used to provide a quantitative basis for discussions of deviations from the Ideal Gas Law.

She has also nearly completed a program, "FEF," which tests the validity of the approximation

$$\Delta G = \Delta H^\circ + T\Delta S^\circ \qquad (9)$$

where ΔG is the free energy of a chemical reaction at an arbitrary temperature T and ΔH° and ΔS° are the standard (T = 298°K) changes in enthalpy and entropy respectively. The test is through comparison with the more accurate

$$\Delta G = \Delta H(T) + T\Delta S(T) \qquad (10)$$

where heat capacity data are used to evaluate $\Delta H(T)$ and $\Delta S(T)$ explicitly for the given temperature T. A table of heat capacity coefficients is stored in the program.

Another student is working on a program incorporating van der Waals' law and the criterion of inherent stability to generate P vs. V isotherms to demonstrate the phenomenon of condensation of a gas to the liquid state.

d. Chemical Kinetics

Another student is starting a project to simulate the time development of chemical reactions for a variety of reaction mechanisms. He has already written a program for the one-step, first-order, irreversible reaction

$$A \xrightarrow{k_a} B \qquad (11)$$

and is working on the consecutive two step reaction

$$A \xrightarrow{k_a} B \xrightarrow{k_b} C \qquad (12)$$

where both reactions are first order and irreversible.

V. Program Distribution

"DIATH2" is currently available through the Quantum Chemistry Program Exchange, Chemistry Department, Room 204, Indiana University, Bloomington, Indiana 47401. As the remainder of programs become ready for distribution, they will also be available from QCPE.

VI. Acknowledgments

To Dr. Norman E. Sondak and the Worcester Area College Computation Center for furnishing computer time and advice on specific programming problems.

To the following WPI undergraduates for assisting on this project: David Lyons, Janet Merrill, Robert Schlain, Benjamin Thompson, and Christopher Williams.

Footnotes:

1. These formulas can be found in any standard introductory quantum chemistry text. For example: L. Pauling and E. B. Wilson, Jr., "Introduction to Quantum Mechanics," McGraw-Hill Book Co., New York, 1935. pp. 326-353.

2. The specific formulas are not repeated here since they are readily available.

3. This discussion is limited to the case of the monobasic weak acid HA. However the program algorithm is capable of handling H_2A and H_3A as well.

Literature Cited

1. (a) W. Heitler and F. London, Z. Physik, 44, 455 (1927).
   (b) Y. Sugiura, Z. Physik, 45, 484 (1927).
   (c) S. C. Wang, Phys. Rev., 31, 579 (1928).

2. (a) L. Pauling, Chem. Rev., 5, 173 (1928).
   (b) B. N. Finkelstein and G. E. Horowitz, Z. Physik, 48, 118 (1928).

3. C. A. Coulson, Trans. Farad. Soc., 33, 1479 (1937).

4. S. Weinbaum, J. Chem. Phys., 1, 593 (1933).

5. M. D. Newton, F. P. Boer, and W. N. Lipscomb, J. Am. Chem. Soc., 88, 2353 (1966).

THEORETICAL CHEMISTRY

by

Lawrence C. Snyder

Bell Laboratories

For years Theoretical Chemistry has existed as two distinct disciplines; Quantum Chemistry which has been mainly concerned with the electronic structure of isolated molecules in their ground and excited states, and Statistical Mechanics which has developed sophisticated ways to deal with the macroscopic properties of ensembles of weekly interacting molecules. The members of these disciplines have had rather little to say to one another, even though the molecular interaction potentials required for the theories of statistical mechanics were almost sure to be generated by the Quantum Chemists.

The Theoretical Chemistry session exhibits the first stages of a joining of these disciplines which will initiate a fruitful new age of Theoretically Chemistry. The member role of the computer in making this synergistic match will be apparent from these talks.

COMPUTERS AND THE SCHRODINGER EQUATION AS A RESEARCH INSTRUMENT AND A TEACHING TOOL

Arnold C. Wahl, Chemistry Division
Argonne National Laboratory, Argonne, Illinois, 60439

Introduction

There exists a wide spectrum of computer applications in chemistry research and teaching as evidenced by this conference. These applications include sophisticated interactive automation of and data processing for experiments[1,2], the exploration of organic synthesis[3], semiempirical[4] and *a priori*[5] theoretical calculations, in addition to a wide variety of simulation studies in chemistry and chemical engineering[6]. In many ways these computing systems can be used as or as part of a research or engineering instrument or as a teaching tool. One of the most intriguing and prophetic applications of the computer which has developed dramatically over the past 10 years is in the *a priori* modeling of atomic and molecular structure and fundamental chemical processes. The reason that this area is exciting is that it provides a means for obtaining chemical information independent from experiment. Further, if reliable atomic and molecular characteristics can be obtained from such *a priori* theory they then can be fed into the mathematical machinery of thermodynamics, molecular dynamics and statistical mechanics to yield the bulk properties of matter, again independent from experiment. We must, of course, be very cautious in making such claims and our ability to make them at present rests ultimately on our facility in approximately solving the Schrodinger equation for groups of atoms and molecules. In the next section we will briefly discuss this capability and then illustrate how it can be *conveniently* used in research and teaching.

Various *a priori* Methods for Approximately Solving the Schrodinger Equation for Atoms and Molecules

A wide variety of methods have been developed both formally and computationally for obtaining molecular wavefunctions and properties.[7-29] There exist both variational , semi-variational and non-variational schemes. In accessing the utility of these methods it becomes clear that they are in general fairly costly and must be pushed close to their "engineering" limits in order to reliably obtain interaction and excitation energies within ± .lev and other properties within 5%. Often this or higher accuracy is needed if even qualitative contact is to be made with modern experiments. It is also clear that except in special cases, of which there are many examples, the Hartree Fock model cannot[30-31] yield chemically accurate potential curves and surfaces due to its two shortcomings. 1) The constrained form of the molecular orbital picture which forces dissociation of the system to improper asymptotic fragments and, 2) the increase in correlation energy associated with molecular formation.

Although wavefunctions of nearly Hartree Fock[32a] quality have been computed for surfaces and large molecules for some time , it has only been recently that methods beyond the Hartree Fock level have yielded useful precision (except for H_3) in potential curves and surfaces.[32b] Another recent advance which opens many previously inachievable areas of chemical interest to *a priori* techniques is the apparent ability to evaluate Van der Waals interaction with a precision of about 10% and possibly better.[33,34]

In summary a combination of Hartree Fock, Multiconfiguration Hartree-Fock, Natural orbital, Straight configuration interaction calculations, and pair theory when appropriately used show promise of producing in a systematic way chemical bond strengths, long range interactions, energy surfaces, term values and many molecular properties at a level of accuracy useful to the chemist. Further conservative estimates of future advances in theory, technique and computer technology predict that such calculations will become an everyday tool of the chemist. In the next section I will present some typical examples of such calculations and then discuss how this relatively new quantitative facility can be interfaced utilizing some concepts of computer interviewing and artifical intelligence with the non specialist and perhaps with the student.

Some Examples

All of the above methods, when properly implemented and thoroughly applied are capable of yielding chemically accurate values of many atomic and molecular properties as well as a good description of the interaction energy between atoms and molecules.

I would now like to briefly review the method and techniques[25] (Optimized Valence Configurations) which we have been developing in our laboratory and illustrate their application to three areas: 1) Chemical bonding, 2) Van der Waal interactions, 3) Chemical reactions.

The essence of our scheme named, "Optimized Valence Configurations" is to take into account only those changes taking place as the molecule forms.[26] This goal is achieved by 1) including those configurations necessary for proper dissociation of the molecule to Hartree Fock atoms, 2) including all configurations which exist in the molecule but vanish formally in the dissociated atoms, 3) including configurations which describe mainly atomic correlation but whose contribution changes within a numerically defined threshold of significance as the molecule forms. An important feature of our method is that we use the multiconfigurational SCF (MCSCF) technique to form all orbitals occurring in all configurations. This has the very important feature of reducing the number of necessary configurations drastically, since each configuration is made as effective as possible. The first two classes of configurations are unambiguous, relatively few in number, can be identified by purely formal considerations, and yield a potential curve which is accurate to within ± .3 eV. The third class must be selected by actual computation and are required for more precision (± .1 eV).

Chemical Bonding

The OVC scheme has been applied to a variety of molecules and bond types and is yielding adequate precision in describing molecular potential curves.[26,35]

When all the important correlation-effects within and between various molecular shells are investigated, it is found that consistent with the philosophy of the OVC method these effects do indeed fall into two fairly distinct categories: the molecular correlation, which vanishes when the molecule dissociates into the constituent atoms, and the atomic correlation, which passes asymptotically to that for the atoms. The very weak interdependence or "coupling" of the two correlation types results into considerable computational simplification by allowing one in the spirit of pair energy to perform MCSCF calculations on groups consisting of a comparatively small number of configurations and to sum their respective contributions to the bonding interaction. The calculations on F_2 yield a potential curve which is very similar to the one derived from experiment. Our calculated[36] values of De, ωe, Re are 1.67, 944, 2.66 compared with the experimental ones of 1.68, 931, 2.68.

Van der Waals Forces

Until recently little progress had been made on the evaluation of Van der Waals forces by a priori variational techniques. Several recent calculations, however, suggest that the following procedure may be adequate for obtaining well depths and position within ten percent[33,37]. 1) Obtain a high precision Hartree-Fock interaction potential; 2) evaluate the intra-atom correlation using only the dominant atomic correlating MCSCF orbitals as a function of the internuclear separation; 3) evaluate the contributions from the dispersion terms. In step (2) it is necessary to include the terms that represent enhanced correlation at a particular atom owing to any significant "transfer of charge" from the other center in the Hartree-Fock approximation.

Such partitioning can result in a great deal of problem simplification. Since each step in this procedure involves only a few configurations to be optimized by the MCSCF process as opposed to a much larger number if all affects must be computed simultaneously. Further calculations indicate that even higher precision may be obtainable by this prescription.

Chemical Reaction Surfaces

We have applied the techniques found successful for evaluating molecular potential curves and long range forces to the calculations of the energy surface for the endothermic chemical reaction[38] $Li + H_2 \rightleftarrows LiH + H$. These calculations are still in progress, however, some interesting features have already emerged. From the nature of the energy surface $E(R_H, R_{Li}, \theta)$ and the changing form of the wavefunction it is clear that reaction only takes place if the H_2 bond is stretched to about 2.5 bohrs. When the dynamics involved in a lithium atom colliding with a hydrogen molecule are considered the necessity for stretching the H_2 bond implies that only highly vibrationally excited H_2 will react with Li.

Automation of *a prori* Techniques

Now that *a priori* techniques exist, which when carefully applied can yield reliable and independent information about simple and fundamental chemical processes (ie. bonding, excitation and ionization, collision, chemical reaction, vibration), it is important to consider how such mathematical modeling can be made readily accessible to the non theoretical chemical researcher and perhaps more importantly to the chemistry student. It is not unrealistic to imagine that much conceptional chemistry on the atomic and molecular level could be greatly enriched if these techniques were available to the student.

In our laboratory we have been developing our computing systems with this communicatability in mind.[39] Our BISON system is based on the following criteria:

(1) Responsiveness to questions relevant to the non-specialized user posed in his language.

(2) Reliability or, at least, a limit of error given in the computed result.

(3) Fool-proof black-box operation in which the computing system makes as many correct procedural decisions as possible in answering a particular chemical question. Implicit in the concept of being responsive to a truly chemical question is that built into this new computing instrument must be a great deal of the specialist's experience. (However, only that experience which has been demonstrated to yield reliable results should be built into the instrument and therefore unproven methods should not be included in such an instrument.)

The instrument that we have designed is being built around BISON, a research system which consists of a self-consistent field program for large diatomic molecules, a properties program for the same systems, a graphical display program which automatically plots charge densities, a transition probability program, a spectroscopic constants program, and a crystal field integrals program, all unified into a well-coordinated operating system which permits the user to do any or all of the above tasks.

This design can be broken down into the following components. (1) The INTERVIEW MODULE of BISON converses with the requestor to define his request more clearly, to inform him of the computer time involved, and to provide him with an estimate of the precision of the answer; (2) the PROCEDURAL-EXECUTIVE MODULE executes a sequence of calculations to answer this chemical question; (3) the COMPUTATIONAL MODULES are used by Procedure Executive to answer the chemical question. The system is currently capable of producing Hartree-Fock wavefunctions and properties for heteropolar or homopolar closed and open shell diatomic molecules containing atoms from rows 1 through 5 of the periodic table. A wide variety of excited states are permitted. It will also produce wavefunctions based on the Adams localized orbital procedure, as well as solve the SCF equation in the presence of an external point charge environment (all necessary integrals are computed); (4) MOLECULAR DISPLAY MODULE. BISON will generate and plot contour diagrams of the orbital and total densities of the molecule. An interesting feature of this process is that the contours are hunted out and plotted automatically and appear on 35-mm film as computer output or on an off-line CalComp plotter. This display feature is particularly intriguing when used to watch a molecule[40] form or to make comparison between a molecule in several states[5], or between molecules of a homologous series[41]. This display feature can be used with Hartree-Fock or multiconfiguration wavefunctions.

It is capable of producing by *a priori* computations, electronic energies, potential curves, charge densities, spectroscopic constants, dissociation energies, many properties dependent in the charge density such as dipole and higher moments, ionization potentials and bond lengths.

The *Procedure Executive* and *Computational modules* have been adequately described previously and we will not discuss them here, however, we have been changing and redesigning the Interview Module and further developing the molecular display capability. In what follows I will present some of the new aspects of interviewing the BISON under which are of general interest in computer interviewing and in the last section will disucss some features of and recent material produced by *Molecular Display*.

The Interview Module

The latest version[42] of INTERVIEW has been designed to use Argonne's time sharing system RESCUE[43]. This version is in a higher level language with good facilities for string manipulation[44]. The user interface of this version was designed to provide considerably more flexibility than the older version[39].

After some experience with our original interview program it became clear that a wider spectrum of users must be accomodated, the interviewing system must be flexible enough to be convenient to a user acquainted with the nature of the data, while having the potential to explain what data is necessary, what form it can take, and what can be done with it, to a user who is in need of instruction in these areas. The system must also be self-explanatory, so little or no preliminary instruction is necessary for the beginner. However, it is undesirable for the frequent user to be forced to tolerate informational messages and explanations regarding aspects of the system already familiar to him.

In its most active form, then, the processor should function as a teaching aid, while at its most passive, it should be a research tool.

To accomodate users in both classes, four distinct levels of interaction are designed into the present interview system, ranging from full prompting and explanations to only a single message. We can group these levels for purposes of explanation into two modes: novice and expert. Within each of these modes, the messages can be abbreviated giving rise to four modes. This basic distinction grew out of an analysis of conversation. In examining conversations, one finds that, at a given instant, one of the participants has control of the direction of the conversation (not necessarily the present speaker). This concept of conversational control has been adopted in the use of these modes. The expert mode allows the user to enter data and make commands in any order and at any time he wishes. The novice mode allows the computer to choose the area of discussion, leading through various topics in an orderly manner, developing a data set with the user, while explaining what can be done with the data and how it is done. Through use of an abbreviation mechanism, the depth of explanation can be varied. Along with this concept of conversational mode, however, comes a unique problem. How can a user change from one mode to another, with a minimum of effort, once he has entered a particular mode? This problem of ultimate control of the conversation, which, of course, must remain with the user, is solved by implementing a command for transfer from the expert to the novice mode (namely 'NOVICE'), but allowing a transfer in the opposite direction implicitly, by ignoring the question asked by the processor and entering a command legal in the expert mode. Initially, the user is asked to rate his expertise on a scale from 0 to 3, and is automatically entered into the appropriate mode. Design of the user interface attempted to follow the user engineering principles discussed by Hanson[45].

There are no specific commands in the novice mode; only responses to the questions asked are processed, and errors in those responses (indicating a lack of understanding) trigger further explanation of the request. The expert mode, on the other hand, contains commands which can be grouped into two categories, data commands and computation commands. A data command is a data name (there exists a flexible, but formally defined list of legal data names), followed by a space, followed by a value, optionally followed by a specification of units and/or a specification of states. All irrelevant information is ignored, such as specification of units in certain cases. Computation commands are requests for the calculation of some chemical property. Such commands are simply passed on to BISON in the appropriate form. In addition to these commands are systems commands, which can be invoked at any point in the program, and which are either commands known to the rescue system or directions to the processor, a few of which are described below.

If the user enters the command 'HELP', operation of the processor is temporarily suspended, and the 'help' processor is entered. The 'help' processor will allow the user to access information regarding such things as processor structure, legal commands, their meaning and usage, and other generally comforting explanations. This is a standard part of all processors in the Rescue system.

'CHECK' is a command useful in the expert mode. It examines whatever data is presently being created, and informs the user of any logical inconsistencies or insufficiencies. The user can then anticipate certain errors before costly computing time is wasted. This function is built into the novice mode.

'FINISH' performs first the 'check' operation, and if no problems are detected, writes a Rescue file with the created data, which can then be submitted to BISON.

'LIST' lists the data so that it can be scanned visually for errors which cannot be detected by the program, such as an incorrect value or a typographical error.

In addition, there are several commands which calculate such things as the cost of calculation, the accuracy of results, and the amount of computing time necessary.

All commands can be prefaced by the keyword 'NO', which will perform the opposite of the operation normally indicated by the following command. For example, the input line 'NO IONIZATION POTENTIAL' would counter any previous command to calculate the ionization potential. When the command 'NO' is irrelevant, it is ignored. Since commands are recognized by scanning a table of keywords, they may be specified by the minimum number of letters (consecutive from the beginning of the word) necessary to distinguish them from other keywords. In most cases, this number is not greater than three, and often one letter is sufficient. Thus, in the above example, 'NO IO' would have been sufficient. Some commands require the user to specify the relevant state of the molecule. During execution, the system remembers a set of one or more current active states. These apply by default if no state is specified for a command which requires one. The default is changed when a user specifies a set of states in a command, or he enters a 'STATE' command. In this way, data and computation commands which are common to more than one state need not be entered a large number of times.

Using these commands and the conversational structure, the processor is capable of both creating a consistent data set suitable for BISON, and serving to explain the function and form of the data to a user who is unfamiliar with the potentialities of the BISON system.

Molecular Display[5,46]

One of the most intriguing, novel and experimentally inaccessible features of the *a priori* mathematical modeling of chemical processes is its ability to yield accurate electronic charge densities and their change at arbitrary magnification for an atom, (Chart 1), molecule (Charts 3 and 4) or during a process such as molecular formation (Chart 2 and Films), ionization, vibration, or chemical reaction (Chart 5). Currently the BISON system plots requested contour values of electronic charge density on a calcomp plotter on 5 millimeter film from the computed wavefunction. Also, one may plot interactively on the 2250 utilizing a precomputed library of atomic and molecular wavefunctions. The interactive system can be used to carefully and quickly design and lay out sequences and charts for educational use. Such computer produced plots have been used in a series of educational films[40] and wall charts[41]. Further such films and charts covering different aspects of atomic and molecular structure and chemical processes are being developed as the accurate wavefunctions describing the system are computed. We are currently experimenting with alternative ways of presenting wavefunctions and electronic charge densities. Among these are clouds of perspective dots, shape diagrams (a surface of constant value), a raster color TV system driven by a stand alone computer[47].

It remains, however, a challenge to strike a balance between asthetic appeal and quantitative information content.

Workshop in Chemistry from Computers

We have used the BISON system and the above concepts in a "Chemistry from Computers" workshop held at Argonne, 1970 and 1971. This workshop involved a series of introductory lectures and then use of BISON by the participants on an actual problem. The participants consisted of college faculty and were not theoretical chemists. Two projects were completed.

1) To evaluate the dissociation energy and various properties for the ground state of the LiH problem utilizing the OVC method.

2) To present qualitative and quantitative evidence as to why the chemical bond in Li_2^+ is stronger than the bond in Li_2 although a bonding electron has been removed. This was accomplished through the visual analysis of charge densities and difference densities and also on the basis of various contributions to the total energy of the systems.

The participant response to this workshop was enthusiastic and it was encouraging that in 1 week with properly designed tools non-specialists could successfully attack on an operational level complex quantum mechanical calculations.

Summary

The relatively new ability of *a priori* techniques for approximately solving the Schrodinger equation to an accuracy which is capable of providing reliable chemical information independent from experiment stimulates the development of effective means of interfacing these techniques with the non-specialist and the student. When such interfaces are coupled with the ability to display the details of simple chemical processes on the atomic and molecular level, a useful teaching aid emerges which could replace or complement laboratory experiments for both student and working chemist.

References

1. P. E. Day and H. Krejci, Proceedings of Fall Joint Computer Conference, page 1187 (1968).

2. J. W. Fracer, Chemical Instrumentation, 2(3), 271 (1970).

3. R. M. Laska, Computer Decisions, p. 39, April 1970 based on W. Todd Wipke's work at Princeton Univ. See also, E. J. Corey and W. T. Wipke, Science, Oct. 10, 1969.

4. J. A. Pople and D. L. Beveridge, Approximate Molecular Orbital Theory, McGraw-Hill (1970).

5. A. C. Wahl, Scientific American, 222, 54 (1970).

6. See Proceedings of Summer Computer Simulation Conference for 1970 and 1971.

7. P. Sutton, P. Bertoncini, G. Das, T. L. Gilbert and A. C. Wahl, I.J.Q.C. IIIS, 479 (1970): A brief review of methods is given in this paper.

8. C. C. J. Roothaan, Rev. Mod. Phys., 23, 69 (1951).

9. W. Kolos and L. Wolniewicz, Phys. Rev. Letters, 20, 243 (1968).

10. W. Kolos, Int. J. Quantum Chem., II, 471 (1968).

11. J. C. Browne and F. A. Matsen, Phys. Rev., 135, A1227 (1964).

12. R. E. Brown and H. Shull, Int. J. Quantum Chem., 2, 663 (1968).

13. F. Grimaldi, J. Chem. Phys., 43, S59 (1965).

14. F. E. Harris and H. H. Michels, Int. J. Quantum Chem., 1S, 329 (1967).

15. D. D. Ebbing and R. C. Henderson, J. Chem. Phys., 42, 2225 (1965).

16. C. F. Bender and E. R. Davidson, J. Phys. Chem. 70, 2675 (1966).

17. C. F. Bender and E. R. Davidson, J. Chem.Phys. 47, 360 (1967).

18. S. Rothenberg, Int. J. Quantum Chem., 1S, 475 (1967).

19. R. Ahlrichs and W. Kutzelnigg, J. Chem. Phys., 48, 1819 (1968).

20. C. Edmiston and M. Krauss, J. Chem. Phys., 45, 1833 (1966).

21. R. Pauncz, Alternate Molecular Orbital Method, Saunders, Philadelphia, 1967.

22. E. Brandas, Int. J. Quantum Chem., 2, 793 (1968).

23. H. Conroy, J. Chem. Phys., 41, 1327, 1331, 1336, 1341 (1964).

24. R. L. Matcha and W. Byers Brown, J. Chem. Phys., 48, 74 (1968).

25. W. Goddard, Phys. Rev., 157, 81 (1967).

26. A. C. Wahl and G. Das, Advances in Quantum Chemistry, 5 (1970).

27. D. F. Tuan and O. Sinanoglu, J. Chem. Phys., 41, 2677 (1964); see also for a general discussion of pair energies, O. Sinanoglu, Advan. Chem. Phys., 2, (1964).

28. R. K. Nesbet, Phys. Rev., 155, 51 (1967).

29. N. C. Handy and S. F. Boys, Proc. Roy. Soc., A311, (1969); A310, 63 (1969).

30. A. C. Wahl, P. J. Bertoncini, G. Das and T. L. Gilbert, I.J.Q.C. 1S, 123 (1967).

31. R. K. Nesbet, Advances in Quantum Chemistry, 3, (1967).

32a. M. Krauss, NBS Technical Note 438 (1967).

32b. M. Krauss, Annual Review of Physical Chemistry, 21, 39 (1970).

33. P. J. Bertoncini and A. C. Wahl, Phys. Rev. Letters, 25, 991 (1970).

34. H. F. Schaefer, O. McLoughlin, F. Harris and B. J. Alder, Phys. Rev. Letters, 25, 998 (1970).

35. G. Das and A. C. Wahl, Phys. Rev. Letters, 24, 440 (1970).

36. G. Das and A. C. Wahl, Submitted to J. Chem. Phys.

37. G. Das and A. C. Wahl, Phys. Rev., 00,000 (1971).

38. A. C. Wahl and G. Das, Proceedings of Santa Cruz Conference on Potential Energy Surfaces in Chemistry (1970), W. A. Lester, Editor, IBM Research Lab., San Jose, California.

39. A. C. Wahl, P. J. Bertoncini, K. Kaiser and R. H. Land, I.J.Q.C. IIIS, 499 (1970).

40. A. C. Wahl and U. Blukic, Atoms to Molecules, Film Loop Series, McGraw-Hill (1969). See also J. Chem. Ed., 45, 787 (1968).

41. A. C. Wahl and M. T. Wahl, Atomic and Molecular Structure, 4 Wall Charts, McGraw-Hill (1970). See also A. C. Wahl, Science, 151, 961 (1966).

42. W. J. Hanson, J. Kaplan and A. C. Wahl, Interviewing by Computer for BISON (in Progress).

43. A. F. Joseph and R. L. Logan, 'RESCUE', A Time-Sharing System, Applied Math. Division, Argonne National Laboratory, Argonne, Ill. 1969 (Unpublished).

44. K. W. Dritz, RXPL Reference Manual, Argonne National Laboratory, TM-197, 1970.

45. W. J. Hanson, Creation of Hierarchic Text With a Computer Display, Argonne National Laboratory, Report Number ANL-7818, 1971.

46. The Film Loop Series, "Atoms to Molecules" and the Chart Series, "Atomic and Molecular Structure" will be on display in addition to pictures of a chemical reaction.

47. W. Lidinsky, Applied Mathematics Division,
Argonne National Laboratory, Argonne, Illinois.

MO Theory as a Chemical Tool[1]

Michael J.S. Dewar

Department of Chemistry, The University of Texas at
Austin, Austin, Texas 78712

Introduction

If quantum theory is to be used as a practical aid to organic chemistry by providing quantitative estimates of molecular structure and chemical reactivity, the results must not only be of sufficient accuracy but must also be achieved without undue cost for molecules of reasonable size.

Since the degree of accuracy required is far outside the reach of current ab initio methods, the only hope lies in some semiempirical approach. Either some approximate and inherently inaccurate ab initio treatment may be shown by experiment to give results of sufficient accuracy, due to a fortuitous cancellation of errors, or sufficient accuracy may be achieved by introducing parameters into the treatment. In either case, the treatment must be thoroughly tested by comparison with experiment before any confidence can be placed in its use for predicting chemical behaviour.

A minimum condition that must be met[2] is that the properties of stable molecules should be reproduced with "chemical" accuracy, i.e.

(a) heats of atomization to the order of ±1 kcal/mole;

(b) bond lengths to the order of ±0.01 Å;

(c) bond angles to the order of ±1°;

(d) force constants to the order of 1%.

If the results are to be meaningful it is also necessary to calculate molecular geometries completely; use of assumed or experimental geometries can lead to large (in a chemical sense) errors in the energy for molecules of quite moderate size. The equilibrium geometry can be inferred only by interpolation from calculations for numerous individual configurations, at least several hundred even for a simple molecule such as methylcyclohexane and at least several thousand to define the reaction path and transition state for a reaction. Unless each calculation can be carried out in seconds rather than minutes, hours or days, the cost of the calculation will be prohibitive in relation to its chemical value.

These considerations exclude the "ab initio SCF" approach. Not only are the results too inaccurate but the cost of carrying out the calculations properly, i.e. with optimization of geometries, is wholly excessive and will remain so in the foreseeable future. Artificies which reduce the computation time, e.g. the use of assumed geometries or very limited basis sets, make the errors, which are already too great, even greater.

The only hope for success at the present time therefore lies in the use of a semiempirical treatment, based on a still cruder approximation to speed up the calculations and fortified by the introduction of parameters to increase its accuracy. The success already achieved with a very preliminary treatment of this kind (MINDO/2[2,3]) seems to leave no doubt concerning the feasibility of such an approach.

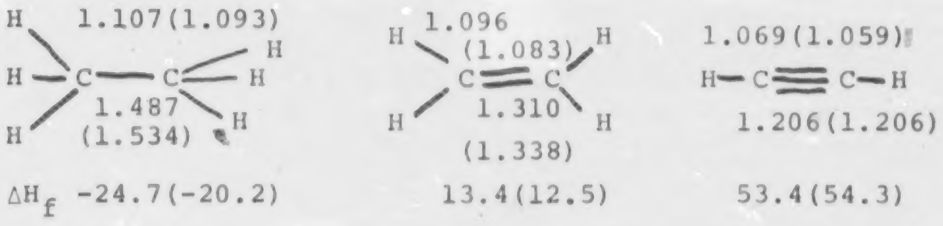
All treatments so far suggested are based on simplified versions of the Roothaan method in which integrals involving diatomic differential overlap are neglected, and also a varying number of integrals involving monocentric overlap (CNDO[4], INDO[5], PNDO[6], NDDO[4]). We have so far mainly concentrated on the INDO[5] approximation, parametrized to give correct heats of atomization and molecular geometries rather than to mimic[5] the results of "ab initio SCF" calculations. To distinguish this approach from the original INDO method, we have termed it MINDO (Modified INDO). The parametrization proved unexpectedly difficult since the parameters are so interrelated that it is impossible to predict intuitively the result of a given variation in any one of them; initially we were able only to reproduce energies of molecules using assumed geometries (PNDO[6], MINDO/1[7]). Eventually we devised[2] a computer program to optimize parameters automatically; use of this led to a treatment (MINDO/2) which gave good estimates of geometries, heats of atomization, force constants, ionization potentials, dipole moments, and nuclear quadrupole coupling constants for a wide variety of molecules containing the combinations of elements CHON, CHF, or CHCl. This initial treatment suffered from a number of deficiencies most of which have been removed in a new version which is almost complete; we have also successfully parametrized the other approximations noted above (CNDO, PNDO, NDDP). We believe that a properly parametrized version of NDDO will provide the optimum solution in terms of accuracy and computation time; at present the computation time is too long by a factor of ten, small enough to be removed by forseeable improvements in computers.[7] The work reported below refers to calculations by MINDO/2. It should be emphasized that its main value lies in the promise of more to come rather than present achievements since we already have better parameters available. Nevertheless the results obtained are already of some chemical interest.

The calculation of molecular geometries has been carried out using a program written by Dr. A. Brown, based on the SIMPLEX method[8]. In order to optimize the geometry of a system defined by $\underline{n}$ independent coordinates, approximately $\underline{n}^2$ calculations are required. This result can also certainly be surpassed. Reaction paths are usually calculated by taking as reaction coordinate some dimension of the reacting system (usually an interatomic distance or angle) which changes monotonically during the reaction and minimizing the energy of the system with respect to the other coordinates for a series of values of the reaction coordinate. As we shall see presently, this procedure, which has the blessing of most standard texts on reaction kinetics, sometimes fails in an interesting and mechanistically significant manner. When it succeeds, the maximum in the plot of energy vs reaction coordinate corresponds of course to the transition state.

Results

Some typical results given by MINDO/2 for various chemical problems are summarized below. For a more detailed review, see ref. (1).

A. Geometries and energies. Comparisons of calculated (observed) bond lengths (Å) and heats of atomization for some simple molecules are shown in Figure 1. The heats of atomization have for convenience been converted to heats of atomization (ΔH_f; kcal/mole at 25°C) using experimental values for heats of formation of atoms. The results are typical for the many molecules we have studied; large deviations occur only in small ring compounds (eg cyclopropane and cyclobutane where the strain energies are underestimated sometimes by as much as 25 kcal/mole) and compounds containing two adjacent heteroatoms (eg peroxides and hydrazines) where the corresponding bond length is too short and ΔH_f much too negative. We feel certain that the former problem can be overcome; the latter, however, is probably an inevitable consequence of the INDO approximation, due to its neglect of certain integrals involving one-center overlap.



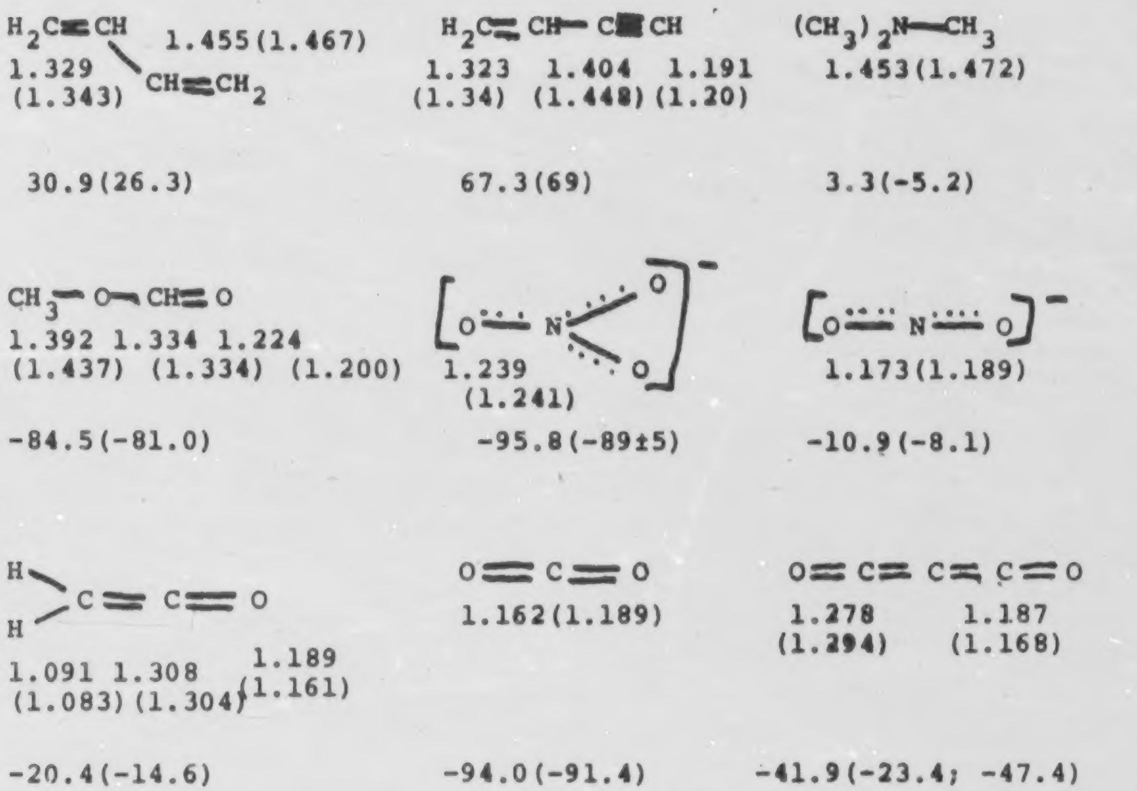

H2C=CH 1.455(1.467)
1.329 (1.343) CH=CH2
30.9(26.3)
H2C=CH—C≡CH
1.323 (1.34) 1.404 (1.448) 1.191 (1.20)
67.3(69)
(CH3)2N—CH3
1.453(1.472)
3.3(-5.2)
CH3—O—CH=O
1.392 1.334 1.224
(1.437) (1.334) (1.200)
-84.5(-81.0)
[O N O O]⁻
1.239 (1.241)
-95.8(-89±5)
[O N O]⁻
1.173(1.189)
-10.9(-8.1)
H H C=C=O
1.091 1.308 1.189
(1.083) (1.304) (1.161)
-20.4(-14.6)
O=C=O
1.162(1.189)
-94.0(-91.4)
O=C=C=C=O
1.278 (1.294) 1.187 (1.168)
-41.9(-23.4; -47.4)

Figure 1. Comparison of calculated(observed) heats of formation (ΔH_f)(kcal/mole) and bond lengths (Å). The "calculated" values of ΔH_f were estimated from MINDO/2 heats of atomization, using experimental values for heats of formation of atoms.

B. Force constants. Table I shows a comparison of calculated and observed force constants for stretching of individual bonds in molecules. The errors are again less by at least an order of magnitude than those given by other methods.

Table I. Calculated (Observed) Stretching Force Constants (mdyne/Å)

Molecule	Bond	Force constant calcd.	Force constant obsd.
CH_4	CH	5.8	5.0
C_2H_6	CC	4.9	4.5
	CH	5.7	4.8
C_2H_4	CC	9.3	9.6
	CH	5.7	5.1

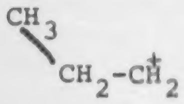
Table I. (continued)

Molecule	Bond	Force constant calcd.	obsd.
C_2H_2	CC	15.1	15.8
	CH	6.1	5.8
H_2O	HO	10.1	7.8
CH_2O	CO	16.5	12.3
NH_3	NH	7.7	6.4
N_2O	NO	14.4	11.4

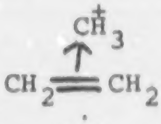
C. Carbonium ions. A comparison[9] of calculated (observed) heats of atomization (as ΔH_f, kcal/mole) for some carbonium ions is shown below. The experimental values are from photoionization studies[10]. Apart from CH_3^+, the agreement is good.

CH_3^+	CH_3CH_2+	$CH_3\overset{+}{C}MCH_3$	$CH_3\overset{+}{C}HCH_2CH_3$	$(CH_3)_3C^+$
276(260)	225(219)	192(190)	183(184)	171(173)

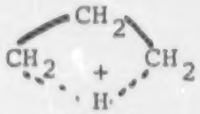
The calculated energies of various isomers of protonated cyclopropane are as follows:



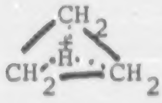
216



195



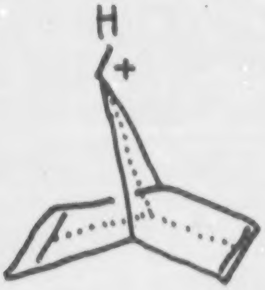
188

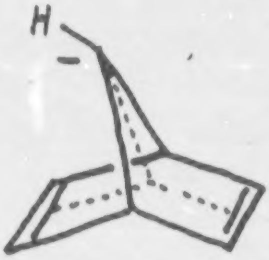


248

The n-propyl cation is predicted to be unstable, isomerizing directly to the edge-protonated isomer. These conclusions differ from those of Pople et al[11], based on ab initio SCF calculations; however they seem to be supported by the available evidence. The solvation energies of n-Pr^+ and i-Pr^+ must be greater than those of the "nonclassical" ions; therefore unless the latter lie well below n-Pr^+ in energy, they would not be formed in solution from n-Pr^+.

The 7-norbornadienyl cation and anion are predicted[12] to have the following unsymmetrical structures; these would be expected on the basis of attractive and repulsive interactions between the C^+ and C^- centers and the double bonds. The barrier to "bridge flipping" in the cation is predicted to be 24 kcal/mole; NMR studies[13] confirm the distorted structure and fix a lower limit of 22 kcal/mole to the "flipping" barrier.





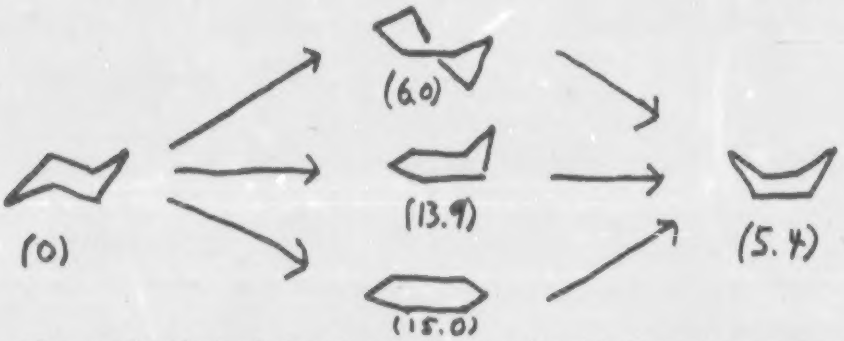
Calculations have also been carried out for a number of other carbonium ions with generally satisfactory results and for related species (e.g. carbenes).

D. Energetics of some simple reactions. Table II compares calculated[2,14] and observed activation energies for twisting about the C=C bonds in ethylene and the cumulenes. The agreement is satisfactory, whereas the "ab initio SCF" method gives values[15] that are too large. A number of other torsional barriers have also been calculated with comparable success.

Table II. Activation energy for Twisting about C=C Bonds.

Molecule	Barrier to rotation (kcal/mole) calcd. MINDO/2	obs.	calcd. ab initio SCF[15]
H_2C	53.5	65.0	138
$H_2C{=}C{=}CH_2$	36.7	-	92
$H_2C{=}C{=}C{=}CH_2$	31.6	30	74
$H_2C{=}C{-}C{-}C{-}CH_2$	24.8	-	-
$H_2C{=}C{=}C{=}C{=}C{=}CH_2$	22.0	20	-

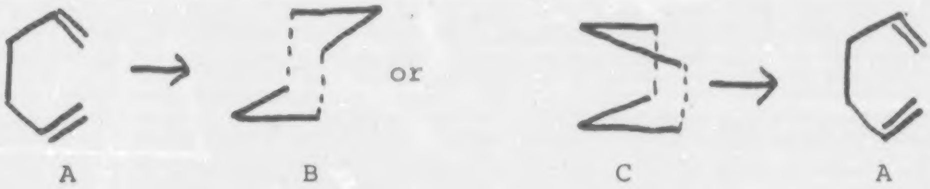
The relative energies (kcal/mole) for the chair and boat conformers of cyclohexane, and of three possible transition states for their interconversion, are as follows:[16]



The calculated heat of isomerization agrees very well with experiment (5.3 kcal/mole). Isomerization is predicted to occur via the first transition state, in agreement with calculations by Hendrickson[17], using the Westheimer-Allinger method. Our value for the activation energy is less than that observed (10.3 kcal/mole[18]) by about the same amount that Hendrickson's value (12.7 kcal/mole) was too high.

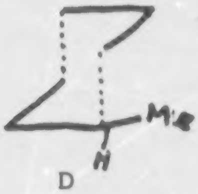
Several other reactions have been studied with comparable success, including hydrogen abstraction from methane by methyl, radical[2], inversion and bond exchange in cyclooctatetraene,[19] carbon atom insertion into double bonds,[20] isomerization of methyl isocyanide to acetonitrile,[14] insertion and addition reactions of singlet and triplet carbene and difluorocarbene,[9,16] and the structure of the Favorskii intermediate.[13]

E. The Cope rearrangement. The Cope rearrangement of 1,5-hexadiene (A) into itself can take place via a chair-(B) or boat-(C) type transition state[21]. The calculated[22] relative energies (kcal/mole) are shown below.

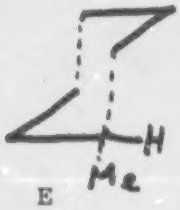


B is correctly[22] predicted to be preferred and the calculated[22] difference in energy between B and C (6.6 kcal/mole) agrees well with that observed[21] (5.7 kcal/mole). Further studies showed that this difference is due to an antibonding interaction between atoms a and b in c, as would be expected from Evans' principle.[23]

Calculations were also carried out for chair transition states D and E with a methyl substituent in a pseudoaxial (D) or pseudoequatorial (E) position. E was predicted to be favoured, in agreement with experiment[24], and the calculated difference in energy (2.0 kcal/mole) agreed well with that (1.5 kcal/mole) observed.[24]



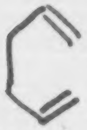
D



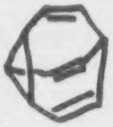
E

These calculations thus reproduced the main results of two extensive experimental studies at a small fraction of the cost.*

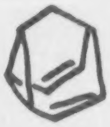
The calculations have been extended[25] to rearrangements in the bullvalene series: as the following results show the calculated activation energies (kcal/mole) agree quite well with experiment (in parentheses).



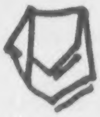
A



F



G



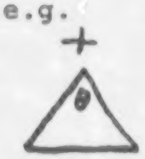
H

The barrier in semibullvalene (H) itself is too small to measure. That in octamethylbullvalene is 6.4 kcal/mole[26]. A detailed study of our MINDO/2 results by energy partitioning[27] indicates, however, that the methyl substituents should increase the barrier to rearrangement in H.

---

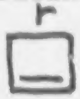
*The estimated cost of the calculations, including salaries, overhead, and cost of computer time at the full commercial rate ($750 per hour) was $5000.

F. Electrocyclic processes. Current studies[28] of the electrocyclic ring openings of cyclopropyl ions and radical to allyl, e.g.


I

J

and of cyclobutene to butadiene,

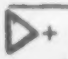
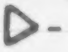
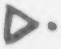

K

L

have led to some interesting conclusions and predictions.

Table III shows calculated activation energies for conrotatory and disrotatory opening of the cyclopropyl systems, the apical angle θ in I being taken as the reaction coordinate. The cation is correctly[29] predicted to open by a disrotatory path and the anion by a conrotatory one, the differences in activation energy being large. Two unexpected features are the prediction that opening of the radical should be disrotatory and almost as stereospecific as that of the cation, and that the reaction of the cation should require activation. The former prediction disagrees with that of Woodward and Hoffmann[30] while the latter seems to be at variance with experimental evidence[31] that concerted rearrangement of I to J assists solvolysis of cyclopropyl esters. The latter discrepancy was resolved by a demonstration that pyramidal cyclopropyl cation, corresponding to the nascent ion in the solvolytic reaction, rearranges without activation by the path shown to be followed by Schleyer et al.. The available evidence[32] seems to support our prediction that I should be a stable species if it is set free in a planar form.

Table III. Calculated[28] Activation Energies for Electrocyclic Reactions of Cyclopropyl

Compound	Calculated activation energy (kcal/mole) disrotatory	conrotatory
 +	7.5	38.0
 -	65.7	30.7
 ·	24.7	52.3

The electrocyclic opening of cyclobutene (K→L) is predicted to take place conrotatorily, in agreement with experiment. In this case the calculations for the "forbidden" disrotatory reaction led to a remarkable result. When the bond length r in K was taken as reaction coordinate, a plot of energy vs r had the unexpected form shown in Figure 2. Neither the forward nor the backward reactions took place! Evidently there are two potential minima for each value of r; one corresponds to a geometry analogous to K, with the CH_2 groups perpendicular to the C_4 unit, while in the other all the atoms are coplanar, as in L.

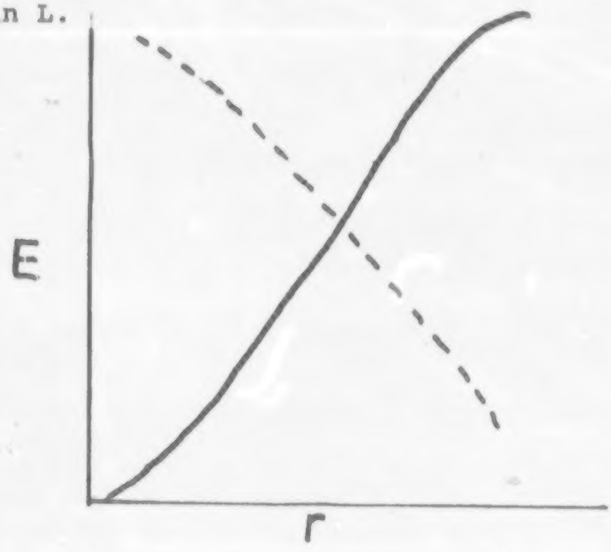


Figure 2. Plots of E vs r for electrocyclic opening of K (—), and closure of L(---).

This result is of course easily explained in terms of Evans' principle[24]. Pericyclic reactions involve a cyclic interchange of bonds, analogous to the interconversion of Kekulé structures in benzene;

If the transition state is aromatic, a hybrid of the two structures is more stable than either so the reaction involves a single potential minimum. If antiaromatic, there are two valleys in the potential surface, separated by a ridge, since intermediate hybrid structures have a higher energy. Similar behaviour should be shown by other reactions involving antiaromatic[24] transition states; this has been so in all cases we have studied (e.g. the "forbidden" reactions in Table III and the cis-cis dimerisation of ethylene to cyclobutane).

Acknowledgement. This work was supported by the Air Force Office of Scientific Research through Contract F44620-70-C-0121 and by the Robert A. Welch Foundation.

References

1) For a review see M.J.S. Dewar, Fortschr. der Chem. Forsch., in press (June 1971).

2) M.J.S. Dewar and E. Haselbach, J.Amer.Chem.Soc., 92, 590 (1970).

3) N. Bodor, M.J.S. Dewar, and E. Haselbach, J.Amer.Chem.Soc., 92, 3854 (1970).

4) J.A. Pople, D.P. Santry, and G.A. Segal, J.Chem.Phys.Suppl., 43, 5129; J.A. Pople and G.A. Segal, ibid. p. 5136.

5) J.A. Pople, D.L. Beveridge, and P.A. Dobosh, J.Chem.Phys. 47, 2026 (1967).

6) M.J.S. Dewar and G. Klopman, J.Amer.Chem.Soc., 89, 3089 (1967).

7) This factor could probably be achieved immediately by a moderate increase in core size and the use of an array processor to accelerate matrix diagonalisation.

8) J.A. Nelder and R. Mead, Computer J., 7, 308 (1964).

9) Unpublished work by Dr. N. Bodor.

10) See J.L. Franklin, J.G. Dillard, H.M. Rosenstock, J.T. Herron, K. Draxl, and F.M. Field, "Ionization Potentials, Appearance Potentials, and Heats of Formation of Gaseous Positive Ions," National Bureau of Standards, Washington, D.C. (1969).

11) L. Radom, J.A. Pople, V. Buss, and P.v.R. Schleyer, *J.Amer.Chem.Soc.*, *93*, 1813 (1971).

12) M.J.S. Dewar and W.W. Schoeller, *Tetrahedron*, in press.

13) M. Brookhart, R.K. Lustgarten, and S. Winstein, *J.Amer.Chem.Soc.*, *89*, 6352 (1967).

14) Unpublished work by Dr. M. Kohn.

15) W.J. Hehre, R. Ditchfield, L. Radom, and J.A. Pople, *J.Amer.Chem.Soc.*, *92*, 4796 (1970).

16) Unpublished work by Dr. W.W. Schoeller.

17) J.B. Hendrickson, *J.Amer.Chem.Soc.*, *83*, 4537 (1961).

18) F.A.L. Anet and A.J.R. Bourn, *J.Amer.Chem.Soc.*, *89*, 760

19) M.J.S. Dewar, A. Harget, and E. Haselbach, *J.Amer.Chem.Soc.*, *91*, 7521 (1969).

20) M.J.S. Dewar, E. Haselbach, and M. Shanshal, *J.Amer.Chem.Soc.*, *92*, 3505 (1970).

21) W. von E. Doering and W.R. Roth, *Tetrahedron*, *18*, 67 (1962).

22) A. Brown, M.J.S. Dewar, and W.W. Schoeller, *J.Amer.Chem.Soc.*, *92*, 5516 (1970).

23) For a summary and references see M.J.S. Dewar, "The Molecular Orbital Theory of Organic Chemistry," McGraw-Hill Book Co. Inc., New York, N.Y. (1969).

24) H.M. Frey and R.K. Solly, *J.Chem.Soc. A*, 1372 (1969).

25) M.J.S. Dewar and W.W. Schoeller, *J.Amer.Chem.Soc.*, *93*, 1481 (1971).

26) F.A.L. Anet, and G.E. Schenck, *Tetrahedron Letters*, *48*, 4237 (1970).

27) M.J.S. Dewar and D.H. Lo, in course of publication.

28) M.J.S. Dewar and S. Kirschner, in course of publication.

29) See R.B. Woodward and R. Hoffmann, *Angew.Chem.Internat.Edit.*, *8*, 781 (1969).

30) R.B. Woodward and R. Hoffmann, J.Amer.Chem.Soc., 87, 39 (1965).

31) P. von R. Schleyer, G.W. van Dine, U. Schöllkopf, and J. Paust, J.Amer.Chem.Soc., 88, 2868 (1966).

32) W. Kirmse and H. Schätte, J.Amer.Chem.Soc., 89, 1284 (1967).

COMPUTER-ASSISTED STUDY OF LIQUID WATER

by

Frank H. Stillinger
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey 07974

On account of its pre-eminent importance to many and diverse branches of science, water has remained the object of intense theoretical study for many years. In the liquid phase, this substance exhibits rather anomalous properties, not the least of which is its apparently unique ability to act as a fluid medium for life processes. In broad terms therefore the thrust of theory consists in showing how the structure of individual water molecules implies anomalous behavior for their aggregates.

Viewed from the modern standpoint, two tasks must be accomplished in sequence. First it is necessary to determine the nature of interactions between water molecules, i.e. the forces and torques that neighbors exert on one another. Subsequently, these interactions must be employed to infer the way that molecules in the liquid arrange themselves next to each other, and the macroscopic properties (density, viscosity, dielectric constant, solvent behavior, etc.) which result from those arrangements. The specific disciplines upon which one must draw for satisfactory quantitative answers in these respective areas are molecular quantum theory, and statistical mechanics.

In the past, theoreticians have not possessed sufficiently powerful tools to achieve comprehensive understanding in either area. As a result, the proposed explanations for the nature of liquid water have been highly speculative and diverse. It is hardly surprising that contradictory pictures have arisen concerning the nature of local molecular order in liquid water. By hindsight, one is inevitably reminded of the famous fable about blind men and the elephant.

Very recently both the molecular quantum mechanics and the statistical mechanics required for a full deductive theory of liquid water have undergone significant advances. These advances have become possible through the wide availability of rapid electronic computers. In the first instance, all-electron Hartree-Fock computations have become feasible for pairs and triplets of water molecules, to permit accurate determination of interaction potentials. Subsequently, molecular dynamics simulations based upon those interactions have been carried out on small samples of liquid water.

Both the quantum-mechanical calculations and the statistical-mechanical simulations have impressive powers of prediction in areas not currently accessible to experiment. One can determine the extent to which the potential is non-additive, and the degree to which that non-additivity depends on hydrogen-bond pattern. In the liquid, it is possible to examine the detailed architecture of hydrogen-bond patterns, and to establish the rate of bond disruption.

The molecular dynamics in particular has a special pedagogical value. Molecular configurations generated during the course of the computer dynamical run have been rendered into the form of stereo photographs which may then be viewed directly. Not only does this visualization capacity demonstrate the inappropriateness of earlier water models based upon disordered crystal structures; it also leads one to formulate novel quantitative questions to pose to the computer.

Thus emboldened by significant advances in understanding pure water, we are now engaged in devising strategies to study simple solutions, and in the longer run to study complicated aqueous systems of biochemical interest. Prospects for major success in the foreseeable future will be discussed.

THE SALSBURG POLYTOPE BOUND ALGORITHM:
APPLICATION TO THE IMPERFECT ANHARMONIC SOLID*

by

C. Grant Miller and Russell D. Larsen
Department of Chemistry
Illinois Institute of Technology
Chicago, Illinois 60616

This manuscript was prepared for presentation at the Conference on Computers in Chemical Education and Research, Northern Illinois University, DeKalb, Illinois, July 1971.

---

*Work supported by Project THEMIS under Contract No. DAAA25-69-C0608

1. Introduction

In elucidating various properties of the classical anharmonic solid (a ν-dimensional lattice system of interacting particles with purely repulsive potentials) the central problem is the solution or approximation of the canonical N-particle configurational integral. A very powerful approach is the use of the cell-cluster scheme which is a systematic procedure for constructing an asymptotic series approximation to the N-body partition function, Q_N [1]. For small N, in the close-packed limit (high density), it is possible to construct regions of configuration space (convex polytopes) the content or νN-dimensional volume of which may then be ascertained. The integration procedure is able to be formulated in such a way that the cell-cluster contributions of any order may be determined exactly for limiting (cum linearized bounds) polytopes. Because of the incredibly large number of pair interaction configurations for N > 5 such an approach to the many-body problem (we require $N = O(10^{23})$) quickly exceeds the capacity of computers currently available. Fortunately, the cell-cluster scheme shows promise of convergence for small N. In fact, convergence has been demonstrated for one-dimensional particles. While this has not been substantiated for higher orders, we presume convergence does exist for 2 and 3 dimensional systems.

Whereas the linearized analysis gives rise to exact results through each order for regular defect-free lattice systems, for imperfect lattices (those that are irregular, distorted and/or contain s-dimensional defects) a curvilinear polytope linearization may still be carried out. However, such an analysis then yields a bound to exact results [2]. It is with such bounds that we deal in this paper for they are the result of considering equilibrium lattice models for fracture and other materials-failure phenomena. We believe that such technologically important solid-state phenomena are many-body cooperative events which may be characterized to first order by classical statistical mechanics using a lattice model with particles interacting via purely-repulsive potentials. We are specifically interested in the properties of imperfect, disordered lattices for which we seek a measure of relative lattice stability. Within the cell-cluster formalism such a measure is directly provided by a change in Helmholtz free energy in going from a perfect lattice to a specific disordered configuration. Alternatively, it is possible to employ the Born stability criteria for various lattice configurations and deformations [3].

For each lattice/sphere system we consider the set of all distinct nearest-neighbor pair interactions, {(n,t)}. Each element of this set is referred to as a cell-cluster of n-particles and topological configuration t. In considering the pairwise interaction of n-particles within a cluster, the remaining

(N - n) particles while constrained to their lattice sites delimit the accessible configurations of the particles within the cluster. In order to ascertain the contribution of a given cluster (of rigid disks of diameter σ, for example) to the free energy it is necessary to evaluate the canonical configurational partition function, $Q_{n,t}$, which may be expressed as:

$$Q_{n,t} = \int \cdots \int_{\mathcal{R}_{n,t}} \prod_{i<j} A(R_{ij} - \sigma) \prod_{i=1}^{n} d\underset{\sim}{R}_i$$

where R_{ij} is the distance between an (ij) pair and the integration is over a specific convex region, $\mathcal{R}_{n,t}$, of configuration space having exact content $\mathcal{P}_{n,t}$. A(x) is the unit step function. The contribution to the Helmholtz free energy, A_N, of a specific cell cluster (n,t) is then:

$$A_{n,t}/k_B T = - \ln \left(Q_{n,t}/n!\Lambda^{2n} \right)$$

where $$\Lambda = h(2\pi m k_B T)^{-1/2}.$$

The single-particle configurational partition function Q_1 for an arbitrary lattice of 2-spheres (disks) is a curvilinear irregular polytope $\mathcal{R}_1$ of content $\mathcal{P}_1$. In the close-packed limit, $V \longrightarrow V_0$, the curvilinear bounds may be replaced by their tangent hyperplanes such that for regular lattices $\lim\limits_{V \longrightarrow V_0} \mathcal{R}_1 = \mathbb{R}_1$ and $\lim\limits_{V \longrightarrow V_0} \mathcal{P}_1 = \mathbb{P}_1$. For irregular lattices, however, $\mathbb{R}_1$ and $\mathbb{P}_1$ are bounds to $\mathcal{R}_1$ and $\mathcal{P}_1$. Similarly for higher-order linearized polytopes, $\mathbb{R}_{n,t}$, the content, $\mathbb{P}_{n,t}$, is a bound to an exact $\mathcal{P}_{n,t}$. For singlet and pair disk configurations and singlet sphere configurations it is possible to calculate the contents $\mathbb{P}_{n,t}$ without computer assistance although with considerable labor. However, in order to carry the cell-cluster scheme to higher order the multiplicity and inordinate complexity of the polytopes precludes a determination of their content were it not for the polytope bound algorithm derived by the late Professor Z. W. Salsburg.

The Salsburg algorithm has a long and rich history. The "homemade" Rice University Computer was designed in part in order to be able to carry out such classical many-body integrations. The pioneering dynamic storage allocation feature of the Rice Computer enabled such integrations involving millions of pair configurations for small N to be effected with a relatively small bulk memory. Due to the vintage of the electronic circuitry (vacuum tubes) the running times for such computations were enormous in comparison to modern standards (70% of a 168 hr. week was not uncommon for certain configurations).

The polytope-bound integration algorithm went through several stages of evolution culminating with a very efficient exponential polynomial scheme [4]. We wish to emphasize that the algorithm effects an analytic integration and not a numerical integration. This is possible because of the nature of the boundary linearization giving rise to step function bounds the arguments of which are polynomials.

We have developed the Salsburg exponential polynomial algorithm in Fortran V for the IIT Univac 1108. (Earlier versions were in the Genie language and Extended Algol.) The scheme has been generalized in order to be able to ascertain the contents of irregular polytopes which arise in the class of problems of interest to us (imperfect, irregular lattices). The algorithm requires the predetermination of three quantities: 1.) The dimensionality of the integral (N) for $Q_{n,t}$, 2.) The number of bounds (K) of the polytope under consideration, 3.) The equations of the polytope bounds (L_i^k).

2. The Salsburg Algorithm

We shall be concerned with the integral representation of the classical canonical partition function for hard-sphere systems. We follow the development of Z. Salsburg, et al. for regular lattices [4] and generalize the prescription to accomodate irregular, imperfect lattice configurations. The canonical partition function for purely-repulsive systems may be expressed as

$$Q_N = \int_{\mathcal{R}} \cdots \int d\underset{\sim}{x}^N P\left(\underset{\sim}{x}^N\right) \prod_{i=1}^{K} A\left(L_i^{(N)}\right) \tag{2.1a}$$

where A(x) is the unit step function which restricts the center of a particle to the interior of a polytope (Voronoi region) defined in terms of its bounding planes which are the arguments of the step function; i.e.,

$$A(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \qquad (2.1b)$$

$$L_i^{(N)} = \sum_{j=1}^{N} a_j(i)x_j + a_o(i) \qquad (2.2)$$

$$\underset{\sim}{x}^N = (x_1, \ldots, x_N)$$

$P(\underset{\sim}{x}^N)$ is a polynomial in the N integration variables. The $a_j(i)$ are vectors (j=1, ..., N) of constants, i=1, ..., k. Eqs. (2.1b) and (2.2) thus define the bounding hyperplanes enclosing an N-dimensional convex polytope $\mathcal{R}$ which is the domain of integration.

The exponential polynomial algorithm is based on the theorem of residues. If we let the polynomial $P(\underset{\sim}{x}^N) = 1$ in Eq. (2.1a) we may write

$$I = \int \cdots \int d\underset{\sim}{x}^N \prod_{j=1}^{K} A\left(L_j^{(N)}\right) \qquad (2.3)$$

or

$$I = \lim_{S\to 0} \int\cdots\int d\underset{\sim}{x}^{N} e^{S\underset{\sim}{\alpha}^{N}\cdot\underset{\sim}{x}^{N}} \prod_{j=1}^{K} A\left(L_{j}^{(N)}\right) \tag{2.4}$$

where the exponential function $e^{y} \rightarrow 1$ as $S \rightarrow 0$ and Eq. (2.4) reduces to Eq. (2.3). $\underset{\sim}{\alpha}^{N}$ is an N vector of arbitrary, nonzero constants and S is a real number.

As in Ref. 4 we note that after integrating over the first i variables we have

$$I = \sum_{q} \int\cdots\int d\underset{\sim}{x}^{N-i} \prod_{j=1}^{K_q} A\left(L_{j,q}^{(N-i)}\right). \tag{2.5}$$

A single term of this sum (the pth) has the form

$$I_p = \int\cdots\int d\underset{\sim}{x}^{N-i} \prod_{j=1}^{K_p} A\left(L_{j,p}^{(N-i)}\right). \tag{2.6}$$

Now in order to integrate such a term by parts over x_{i+1} we define

$$u = \prod_{j=1}^{K} A\left(L_{j,p}^{(N-i)}\right)$$

and

$$dv = e^{S\underset{\sim}{\alpha}^{N} \cdot \underset{\sim}{x}^{N}} dx_{i+1} \, .$$

Also, let

$$x_{i+1} \equiv -\frac{1}{a_{m,i+1}}\left(a_{m,0} + \sum_{k=i+2}^{N} a_{m,k}x_{k}\right).$$

Then,

$$du = \sum_{m=1}^{K} a_{m,i+1}\delta\left(L_{m}^{(N-i)}\right) \cdot \prod_{j=1}^{K} A\left(L_{j}^{(N-i)}\right) dx_{i+1}$$

$$v = \frac{1}{S\alpha_1} e^{S\underset{\sim}{\alpha}^{N-1}\cdot\underset{\sim}{x}^{N-1}} \left(e^{-S\left[a_{m,0}/a_{m,1}\right]} \right)$$

As $\mathcal{R}$ is bounded,

$$uv\Big|_{-\infty}^{\infty} \equiv 0.$$

Hence,

$$I_p = -\lim_{S\to 0} \frac{a_{m,1}}{S\alpha_1} e^{-S\left[a_{m,0}/a_{m,1}\right]\alpha_1} \int\cdots\int d\underset{\sim}{x}^{N-i-1} e^{S\underset{\sim}{\alpha}^{N-1}\cdot\underset{\sim}{x}^{N-1}} dx_{i+1}$$

$$\otimes\, \delta\left(L_{i,p}^{(N-i)}\right) \prod_{j=1}^{K_p} A\left(L_{j,p}^{(N-i)}\right)$$

Integrating over x_{i+1},

$$I_p = -\lim_{S\to 0} \sum_{m=1}^{K_p} \frac{\operatorname{sgn} a_{m,1}}{S\alpha_1} e^{-S\left[a_{m,0}/a_{m,1}\right]\alpha_1}$$

$$\otimes \int\cdots\int d\underset{\sim}{x}^{N-1} e^{S\underset{\sim}{\alpha}_i^{N-1}\cdot\underset{\sim}{x}^{N-1}} \prod_{j=1}^{K} A\left(L_{j,m}^{(N-1)}\right)$$

where $$\operatorname{sgn}(x) = \begin{cases} x/|x|, & x \neq 0 \\ 0 & x = 0 \end{cases}$$

The N-dimensional integral results from an integration over each x_i each of which leaves a set of N-i-1 dimensional integrals. On integrating over N variables the result is

$$I = \lim_{S \to 0} \sum_{\tau} \frac{c_\tau e^{S q_\tau^{(0)} \cdot \alpha^N}}{S^N \prod_{j=1}^{N} \left(q_\tau^{(j)} \cdot \alpha^N \right)} . \tag{2.7}$$

Applying the residue theorem to the limit gives the desired analytic representation of the canonical partition function

$$I = \frac{1}{N!} \sum_{\tau} c_\tau \frac{\left(q_\tau^{(0)} \cdot \alpha^N \right)^N}{\prod_{j=1}^{N} \left(q_\tau^{(j)} \cdot \alpha^N \right)} \tag{2.8}$$

where the $q_\tau^{(n)}$ and c_τ are constants determined by the integration procedure.

3. Illustration of Algorithm: The Niggli Lattice

To illustrate the functional form of this rather obscure final result let us consider the single particle configuration space bound $\mathbb{R}_1$ for the Niggli Lattice illustrated in Fig. 1. It is necessary to specify the dimensionality of the integral ($N = 2$), the number of bounding planes to $\mathbb{R}_1$ ($K = 5$), and the equations of the polytope bounds, $L_i^{(2)}$, $i = 1,\dots, 5$:

$$\begin{aligned} 0 &= 1 + x_1 \\ 0 &= 1 - x_1 \\ 0 &= 1 - x_2 \\ 0 &= 1 + x_1 - x_2 \\ 0 &= 1 - .182x_1 + 1.19x_2 \end{aligned} \tag{2.9}$$

$\underset{\sim}{a}^N\cdot\underset{\sim}{x}^N$ is then an arbitrary N-dimensional function which we choose to be

$$\underset{\sim}{a}^N\cdot\underset{\sim}{x}^N = 1\cdot x_1 + \frac{e}{2.5}\cdot x_2 + \dots + \left(\frac{e}{2.5}\right)^{N-1}\cdot x^N$$

where e is the Naperian base. Thus, Eq. (2.3) becomes

$$\lim_{S\to 0}\iint dx_1 dx_2 e^{S\left(x_1+\frac{e}{2.5}x_2\right)} A(1+x_1)A(1-x_1)A(1-x_2)$$

$$\textcircled{X}\ A(1+x_1-x_2)A(1-.182x_1+1.19x_2) \qquad (2.10)$$

We proceed by integrating Eq. (2.10) over the variable x_2. Consider the five Heaviside functions representing the bounding polytopes in order. If x_2 does not appear explicitly in the expression of the bound (consider bounds 1 and 2 of (2.9)), then the content of that bound is zero, (Eq. 2.1a), and we proceed to the next bound.

Consider bound 3 at Eq. (2.9). Solve that bound for

x_2

$$x_2 = 1$$

Eq. (2.6) states that the contribution of this bound to the integral is given by

$$I = -\lim_{S\to 0} \frac{(-1)}{S\,\frac{e}{2.5}} \int dx_1 e^{S\left(x_1+\frac{e}{2.5}\right)} A(1+x_1)A(1-x_1)A(0)A(x_1)$$

$$\otimes\ A(2.19-.182x_1) + \ldots \qquad (2.11)$$

where the (-1) denotes the sign of the coefficient of x_2 in the bound 3) of Eq. (2.9), $\frac{e}{2.5}$ denotes the coefficient of x_2 in $\alpha^N \cdot x^N$, $e^{S(x_1+\frac{e}{2.5})}$ results from substituting x_2=1 into $\underset{\sim}{\alpha}^N \cdot \underset{\sim}{x}^N$, and the Heaviside functions result from substituting x_2=1 into the previous Heaviside functions of Eq. (2.10), and the +(...) denotes that there will be similar contributions from each of the other polytope bounds of Eq. (2.9). It is worth noting that for some bounds (5 and 6) x_2 can not be expressed as a constant but rather as a function of the other variables ($x_2 = a + bx_1$). Our procedure remains the same in this case as we substitute this expression for x_2 into $\underset{\sim}{a}^{N}\cdot\underset{\sim}{x}^{N}$ and the Heaviside functions.

The Heaviside functions of Eq. (2.11) are redundant. Since there remains only an integration over one variable x_1, x_1 can have only two bounds, upper and lower bounds. We eliminate the extraneous bounds. A(0) may be eliminated since A(c) = 1 where c is a constant (Eq. 2.1a). Fig. 2 illustrates our considerations in eliminating the two other extraneous bounds. We wish to consider only the area of the irregular pentagon of Fig. 2. The bound $A(2.19 - .182x_1)$ is defined by the intersection of bounds 1) and 5) of Eq. (2.9), thus extending beyond the pentagonal polytope and it is eliminated. We are considering the area enclosed by bound 3). This bound extends from $x_1 = +1$ to $x_1 = 0$. Since $A(1 + x_1)$ defines a bound ($x_1 = -1$) outside this region, it must also be an extraneous bound and is eliminated. We are then left with $A(1 - x_1)A(+x_1)$

as our limiting bounds on x_1.

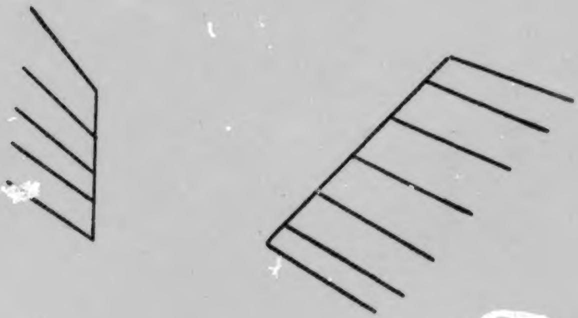
We have presented a strictly qualitative argument for the elimination of extraneous bounds. For a mathematical algorithm it is necessary to develop a quantitative procedure for eliminating such bounds. This may be accomplished by noting that the two nonextraneous bounds form the smallest region of integration for x_1, (from -1 to 0). To eliminate extraneous bounds consider the Heaviside functions of Eq. (2.11). These are of the form $A(a + bx_1)$ where a and b are constants.

If $b = 0$, eliminate the Heaviside function.

If $b > 0$, eliminate all bounds with $b > 0$ except that one with the smallest value of a. Call it lower bound.

If $b < 0$, eliminate all bounds with $b < 0$ except that one with the smallest value of a. Call it upper bound.

There will remain two bounds, one for $b > 0$ and one for $b < 0$ as desired. In all cases where $N \leq 2$, this procedure assures that the integration will be over the proper region. Occasionally extraneous bounds arise due to pair correlations between two or more particles which assume the form



where the hatched area represents the area of integration. Note that the area between the two bounds, in which we are interested, is excluded and hence this configuration represents an extraneous set of bounds which must be eliminated. For such a configuration, the lower bound we have calculated will be greater than the upper bound we have formed. By testing for this property we may eliminate all sets of two bounds, both upper and lower bounds, where lower bound > upper bound.

With these considerations Eq. (2.11) may be written as

$$I = - \lim_{S\rightarrow 0} \frac{-1}{S\,\frac{e}{2.5}} \int dx_1 e^{S\left(x_1 + \frac{e}{2.5}\right)} A(1-x_1)A(+x_1) + \ldots \qquad (2.12)$$

We may now proceed to integrate Eq. (2.12) over the variable x_1 in the same manner as we integrated Eq. (2.10) over x_2. Then:

$$I = \lim_{S\rightarrow 0} \frac{1}{S\,\frac{e}{2.5}} \left[\left(\frac{1}{S(1)} e^{S\left(1 + \frac{e}{2.5}\right)} A(0)A(1) \right) + \left(- \frac{1}{S(1)} e^{S\,\frac{e}{2.5}} \right.\right.$$

$$\left.\left. \textcircled{X} \; A(1)A(0) \right) \right] + \ldots \qquad (2.13)$$

where the quantity inside the first set of ( ) is obtained from considering the bound $A(1-x_1)$ of Eq. (2.12), +1 is (-) the sign of the constant multiplying x_1 in that Heaviside function. S(1) is the constant multiplying x_1 in $\underset{\sim}{\alpha}^{N}\cdot\underset{\sim}{x}^{N'}$;

in Eq. (2.12). $S\left(1 + \frac{e}{2.5}\right)$, A(0), and A(1) result from solving $A(1 - x_1)$ for $x_1 = -1$ and substituting this value into the exponential and Heaviside functions of Eq. (2.12). The quantities inside the second set of ( ) are obtained in like fashion for A(+x) at Eq. (2.12). If we note that A(0) and A(1) = 1, Eq. (2.13) is of the form

$$I = \lim_{S\to 0} \frac{1}{S^2 \frac{e}{2.5}} \left[e^{S\left(1+ \frac{e}{2.5}\right)} - e^{S \frac{e}{2.5}} \right] + \ldots \qquad (2.14)$$

We similarly obtain the results of integrating over the other bounds. These are

$$1.) \Longrightarrow 0$$

$$2.) \Longrightarrow 0$$

$$4.) \Longrightarrow \frac{1}{S^2 \frac{e}{2.5}\left(1 + \frac{e}{2.5}\right)} \left[-e^{-S} + e^{S \frac{e}{2.5}} \right]$$

$$5.) \Longrightarrow \frac{-1}{S^2 \frac{e}{2.5}\left[1 + .153 \frac{e}{2.5}\right]} \left[-e^{-\left(1+ \frac{e}{2.5}\right)} + e^{S\left[1-.69 \frac{e}{2.5}\right]} \right]$$

Eq. (2.14) is now represented as the sum of 1 thru 5). Taking the limit as S→0, according to the residue theorem, we obtain

$$I = \frac{1}{2!}\left[\frac{[1 + c]^2}{c} - c - \frac{[1]^2}{c[1 + c]} + \frac{c}{1 + c} + \right.$$

$$\left. + \frac{(1 + c)^2}{c[1 + .153c]} - \frac{[1 - .69c]^2}{c(1 + .153c)}\right]$$

where $c = \frac{e}{2.5}$. Then I = 3.19.

We have adopted this algorithm to the Univac 1108 computer using Fortran V. A calculation which would take on the order of a month by hand and twenty minutes of computer time for a numical integration takes less than half a minute. In contrast to the original Salsburg algorithm written in Algol, our Fortran version may be used to determine the configurational partition functions of deformed and disordered lattices. The Fortran version has enabled us to model cooperative lattice phenomena within the cell-cluster expansion formalism. For such models we cannot overemphasize the utility of the Salsburg algorithm and this generalized version of it.

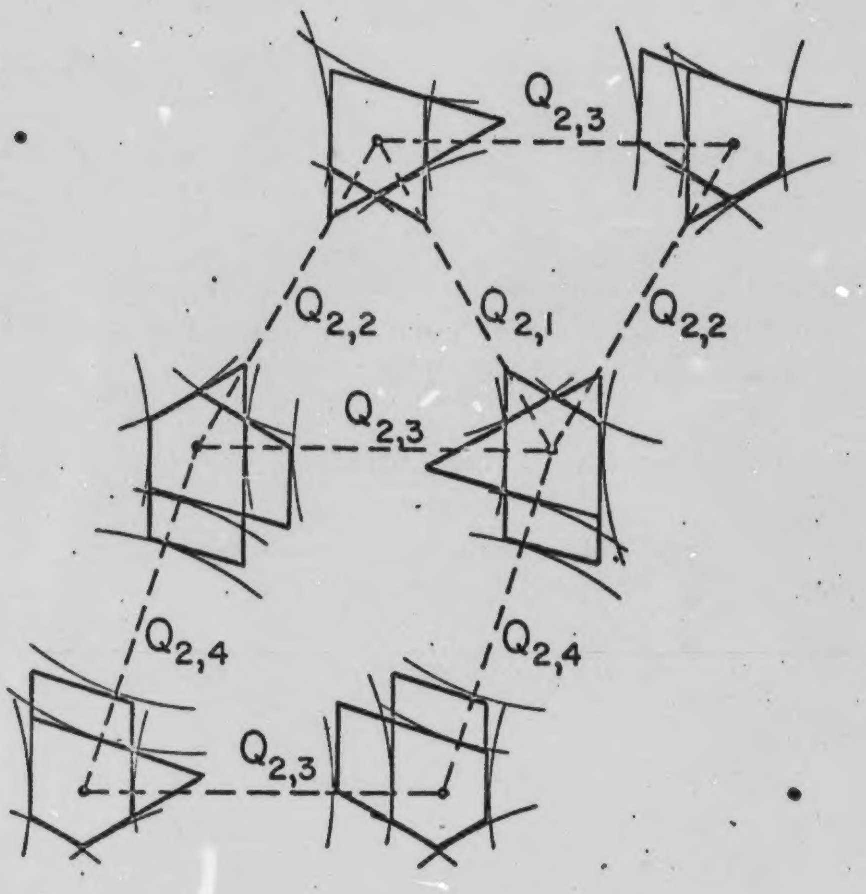
References

1. F. H. Stillinger, Jr. and Z. W. Salsburg, J. Stat. Phys. 1, 179 (1969).

2. G. Miller and R. Larsen, Project THEMIS, Special Technical Report #3, Illinois Institute of Technology, October 1970; to appear in J. Comp. Phys.

3. F. Milstein, Phys. Rev. B, 3, 1130 (1971).

4. W. Rudd, Z. Salsburg, and L. Masinter, J. Comp. Phys. 5, 125 (1970).


$Q_{2,3}$
$Q_{2,2}$
$Q_{2,1}$
$Q_{2,2}$
$Q_{2,3}$
$Q_{2,4}$
$Q_{2,4}$
$Q_{2,3}$

HIGH DENSITY APPROXIMATION TO PAIR SUBFIGURES OF NIGGLI PACKING, I

FIGURE 1

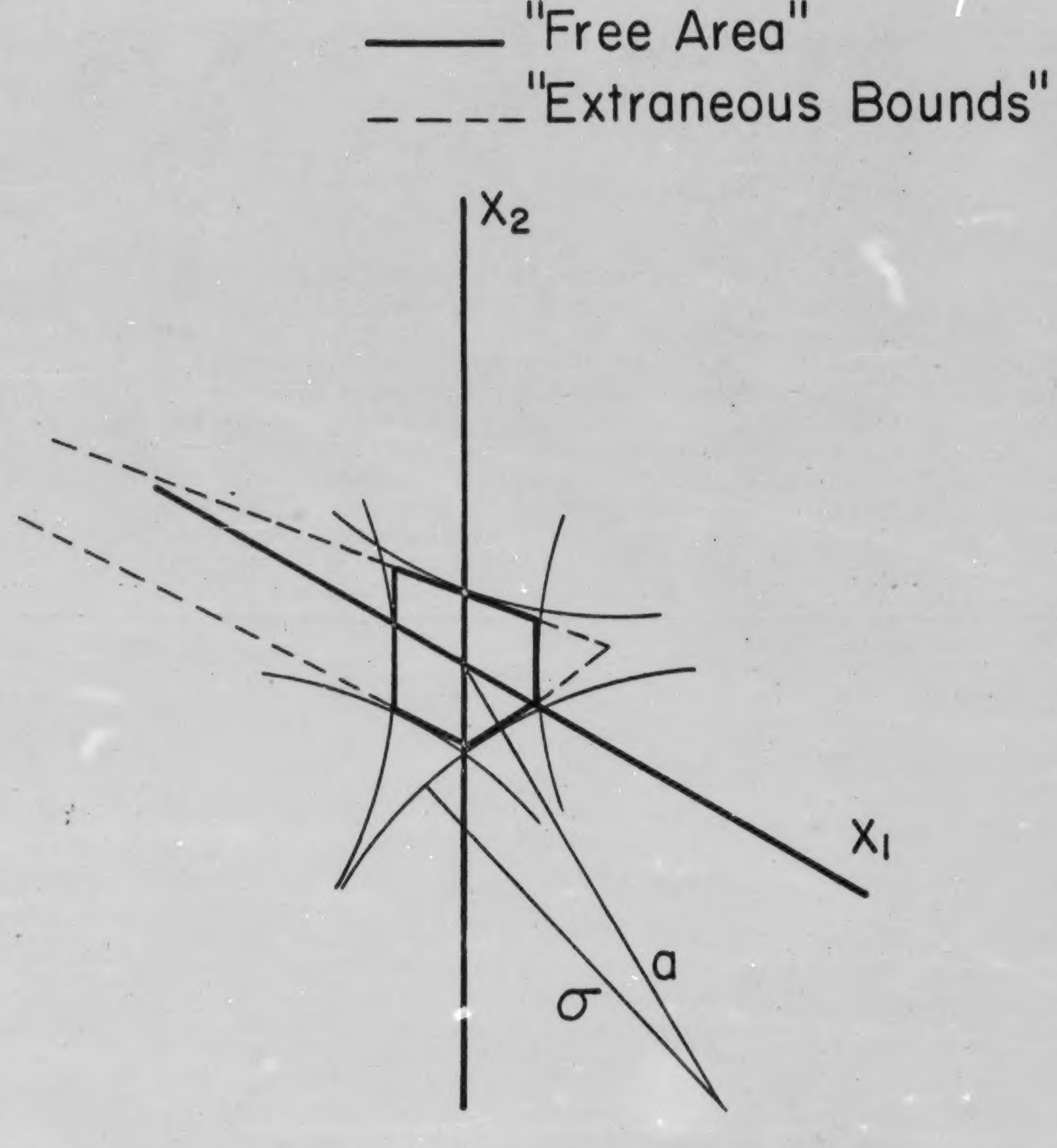

"Free Area"
"Extraneous Bounds"
X_2
X_1
σ
a

Fig.2 NIGGLI I "FREE AREA" and "EXTRANEOUS BOUNDS"

Program: 'SYMMETRY'

Author: Dr. Ronald A. Wohl
School of Chemistry
Rutgers University
New Brunswick, N.J. 08903

Abstract: A computer program module is described which determines the molecular point group of a molecule and prints the corresponding character table. The group character table is used in a form suitable for subsequent use by other program modules which are able to perform a complete symmetry analysis of the molecule, i.e. they can e.g. determine the atomic representation, the structure of the representation (irreducible representations), sets of equivalent atoms, atomic orbitals or internal coordinates and set up symmetry coordinates or symmetry orbitals. These modules may be used in a wide variety of programs for vibrational analysis, molecular orbital theory and electronic spectroscopy, crystal field theory and crystal structure determination.

The concepts of symmetry and group theory have found widespread use in chemistry[1]. Thus the molecular point groups are used in stereochemical arguments[2] (chirality; chemical equivalence of atoms and bonds; presence or absence of a dipole moment), vibrational analysis[3] (symmetry species; degeneracy and activity of fundamentals, overtones, combination bands; selection rules; symmetry coordinates; polarization of Raman bands), molecular orbital theory and electronic spectroscopy[1][4] (species and degeneracy of molecular orbitals, selection rules, vibronic transitions) and crystal field theory[1][5] (crystal field splitting; correlation diagrams; spectral and magnetic properties). The extension from point groups to lattice groups includes applications in the vibrational analysis of crystals (lattice modes; selection rules; site symmetry and factor group analysis), other physical properties of crystals[6] and the determination of the space group and crystal class itself.

Although countless computer programs in the above subject fields have been written very few programs so far have made use of symmetry at all, and no program, to the best of our knowledge, has handled the entire symmetry analysis completely by itself. In those few programs making limited use of symmetry, the symmetry information has usually to be "set up" by hand and entered as data specific for the problem at hand. (E.g. The U-matrix in vibrational analysis for converting internal coordinates to symmetry coordinates; or, the sets of equivalent nuclei in NMR spectra simulation programs). The disadvantages of this procedure are obvious.

It is for this reason, that we have written a program module, that completely automatically without any input other than the masses and coordinates of the atoms determines the molecular point group of a molecule and prints the corresponding character table. The group character table is used in a form suitable for subsequent use by other program modules, which can perform a complete symmetry analysis, such as determine the atomic representation, the structure of the representation, the sets of equivalent atoms and internal coordinates, and set up symmetry coordinates and symmetry molecular orbitals. The extension to the symmetry analysis of crystal lattices is easily feasible and planned for the near future. The three most important subroutines of this program module are briefly discussed below.

Subroutine PNTGR

This subroutine is the central part of the program; it determines all molecular point groups up to order 8 of the main axis of symmetry with the exception of those point groups giving rise to spherical tops, which are handled by the subroutine SPHERE.

A simplified flow-chart of this subroutine is shown in Fig. 1. This flow-chart is based upon the distribution of the various types of symmetry elements in the various classes of point groups, which are summarized in Table 1. Similar procedures and flow-charts for the systematic identification of the point group by visual inspection of a molecular model have been published[7]. The application to a computer program required some modifications. A particular problem not present in the visual inspection is the specific orientation and reorientation of the molecule in the Cartesian coordinate system required during the computer evaluation.

The presence of the various possible symmetry elements is tested for by applying the corresponding transformation matrices to the molecule and checking, whether the transformation corresponds to a symmetry operation. The mathematics required for this part are well known and can be found in all books dealing with molecular symmetry[1a),1b)].

4-special points may be mentioned.

Table 1: Distribution of the symmetry elements in the various types of molecular point groups.

Point Group	I_z	S_n	σ_h	σ_v	C_2'
C_s	1	$(1)^{a)}$	$(1)^{b)}$	$(1)^{b)}$	-
C_n c)	n	-	-	-	-
C_{nh}	n	$(+)^{d)}$	+	-	-
C_{nv}	n	-	-	+	-
S_n e)	n/2	n	-	-	-
D_n	n	-	-	-	+
D_{nh}	n	$(+)^{d)}$	(+)	+	+
D_{nd}	n	$(+)^{d)}$	-	+	+

I_z = order of symmetry element C_n, i.e. order of main rotatory axis (=Z-axis).

σ_h = horizontal plane; σ_v = vertical plane; C_2' = perpendicular 2-fold axis; + means symmetry element is present; - means symmetry element is absent.

a) The symmetry plane is equivalent to S_1. See also b).

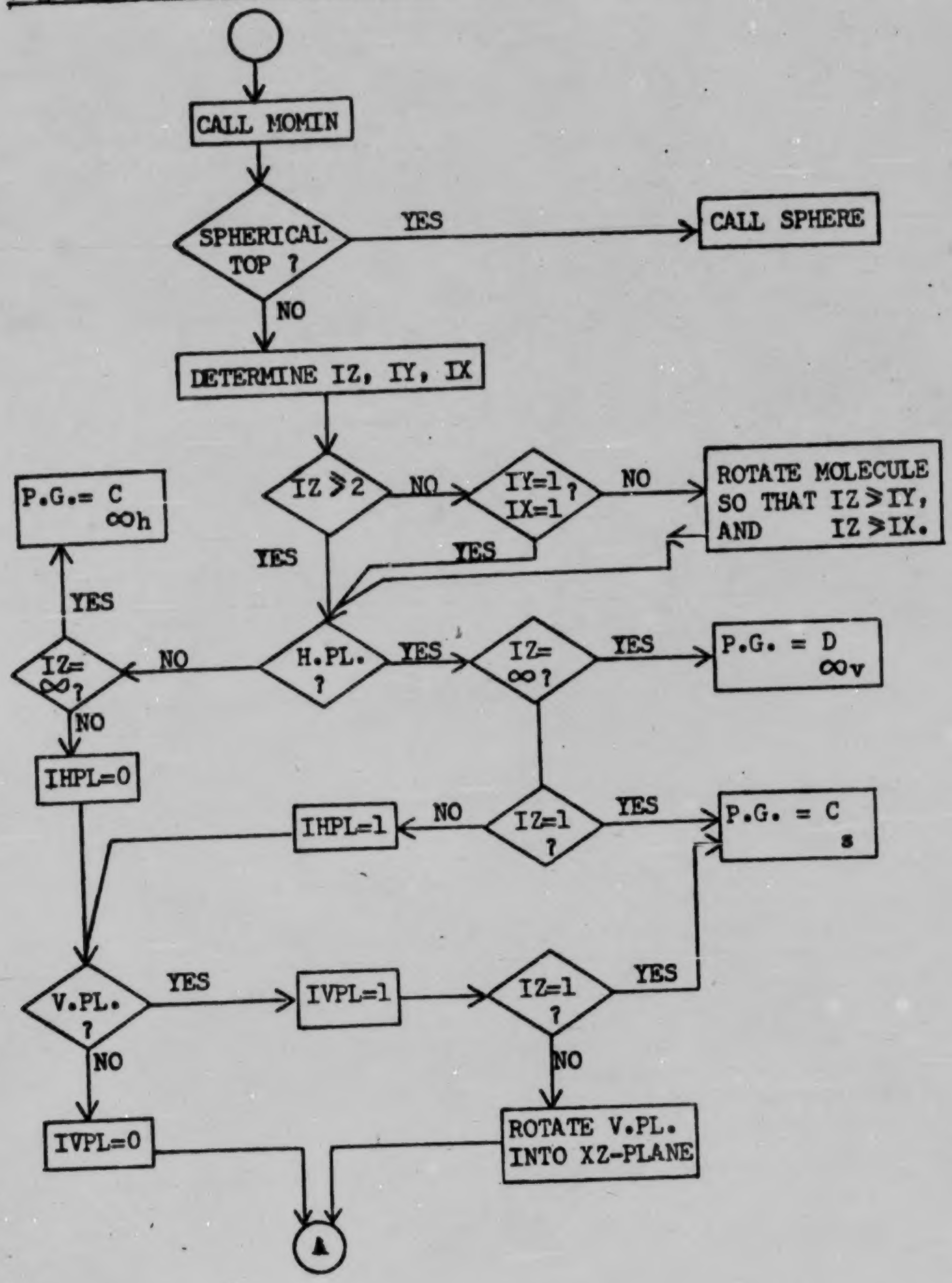
b) The molecule possesses one plane of symmetry. Normally this plane cannot be designated as a horizontal or vertical plane. During the point group evaluation, the molecule can have, however, its symmetry plane oriented either "horizontally" , (i.e. in the xy-plane)or "vertically" (perpendicular to the xy-plane, i.e. through Z-axis).

c) Includes C_1 = molecule contains no symmetry element except the identity.

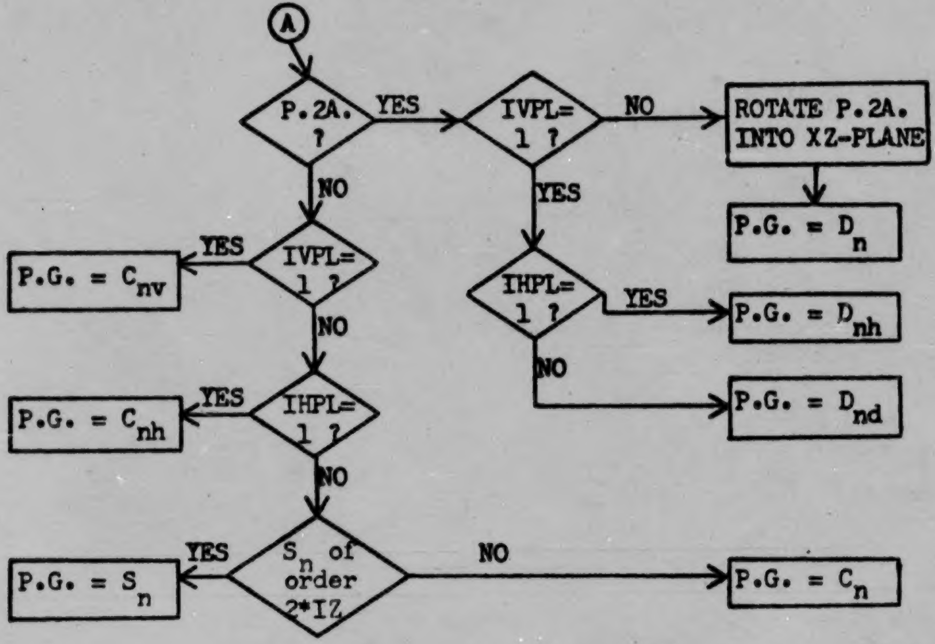
d) This symmetry element is implied by the remaining symmetry elements, but is not tested for during the computation.

e) Includes $S_2 = C_i$.

Fig. 1 : FLOWCHART OF SUBROUTINE PNTGR


CALL MOMIN
SPHERICAL TOP ?
YES
CALL SPHERE
NO
DETERMINE IZ, IY, IX
IZ ⩾ 2
NO
IY=1 IX=1 ?
NO
ROTATE MOLECULE SO THAT IZ ⩾ IY, AND IZ ⩾ IX.
YES
YES
P.G.= C ∞h
YES
IZ= ∞ ?
NO
H.PL. ?
YES
IZ= ∞ ?
YES
P.G. = D ∞v
NO
IHPL=0
IHPL=1
NO
IZ=1 ?
YES
P.G. = C s
V.PL. ?
YES
IVPL=1
IZ=1 ?
YES
NO
NO
IVPL=0
ROTATE V.PL. INTO XZ-PLANE
A

(CONTINUATION OF FIG. 1: FLOWCHART OF SUBROUTINE PNTGR


A
P.2A. ?
YES
NO
IVPL= 1 ?
NO
YES
ROTATE P.2A. INTO XZ-PLANE
P.G. = D_n
IHPL= 1 ?
YES
NO
P.G. = D_{nh}
P.G. = D_{nd}
YES
IVPL= 1 ?
P.G. = C_{nv}
NO
YES
IHPL= 1 ?
P.G. = C_{nh}
NO
YES
S_n of order 2*IZ
P.G. = S_n
NO
P.G. = C_n

LEGEND:

IZ = Order of rotation axis collinear with the z-axis
IY = " " " " " " " y-axis
IX = " " " " " " " x-axis
IHPL = Index, whether horizontal plane of symmetry is present (= 1) or not (= 0).
IVPL = Index, whether vertical plane of symmetry is present (= 1) or not (= 0).
IP2A = Index, whether perpendicular 2-fold axis is present (= 1) or not (= 0).
H.PL. = Horizontal plane of symmetry
V.PL. = Vertical plane of symmetry
P.2A. = Perpendicular 2-fold axis
P.G. = Point group

---

1. The molecule is initially oriented after its principal moments of inertia by the subroutine MOMIN. Symmetric top molecules are oriented such that the main axis of symmetry is collinear with the Z-axis.

2. The order of the rotation axis collinear with the Z-axis is determined,by rotating the molecule around the Z-axis by the angle $2\pi/n$, where n starts with 8 and decreases to 2. Thus to find the correct order of the main symmetry axis it is necessary to test for the highest possible order first.

3. Symmetric top molecules present a special problem similar to that posed by spherical tops (see below) although of less complexity. Namely the molecule cannot be preoriented in the xy-plane because the corresponding two principal moments of inertia are equal. Thus the orientation of possible vertical planes of symmetry and perpendicular 2-fold axes of rotation cannot be known a priori. Therefore all possible pairs of atoms are tested for being both suitable and symmetry equivalent. Suitable in this context means, that both atoms have the same atomic weight and the same Z coordinate. Symmetry equivalent means that a transformation metrix can be found, which transforms atom A into atom B and simultaneously is also a symmetry operation for the molecule as a whole.

4. After the point group has been determined, the molecule will be reoriented in the Cartesian coordinate system, if necessary, to follow all the recommendations given in the Mulliken report[8].

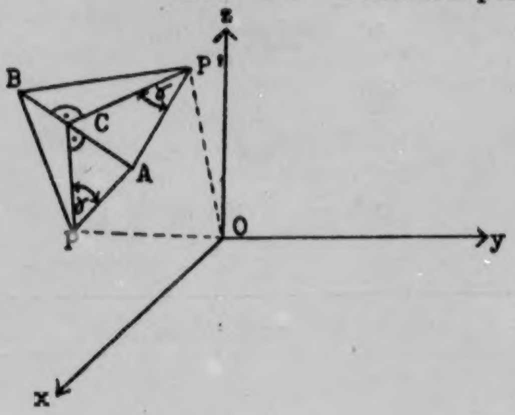
Subroutine SPHERE

This subroutine determines the point group of all spherical tops, i.e. the point groups related to the geometrically regular solids: tetrahedral: T, T_d, T_h; octahedral: O, O_h and icosahedral: I, I_h. (The subroutine also evaluates the rare case of accidentally degenerate spherical tops).

The complete procedure for the determination of the spherical point groups is rather complex. Spherical top molecules cannot be oriented easily after a main axis of symmetry, since the 3 principal moments of inertia are equal. Therefore no symmetry elements oriented a priori in the Cartesian coordinate system can be used for testing the molecule. (Neither must any atom of the molecule lie on a symmetry element). The rather complex procedure necessary then is the following: All possible pairs of atoms of the molecule are tested for being suitable and for being symmetry equivalent. A pair of atoms is suitable if both atoms have the same atomic weight and the same distance from the center of mass. The two atoms are then symmetry equivalent if a transformation matrix can be found, which transforms atom A into atom B, while being at the same time a symmetry operation for the molecule as a whole. Given the two points A and B and the order of the sought rotation axis, the orientation of the axis (of which there are 2) can be calculated by analytical geometry. The corresponding transformation matrix can then be set up and be tested for being a symmetry operation.

Fig. 2 illustrates the geometric construction we have used in order to calculate the orientation of a possible rotation axis of order n, which transforms atom A into atom B.

Fig. 2: Location of points P and P' which define rotation axis of order n to transform point A into point B.



OP defines rotation axis
C = midpoint between A and B
$\gamma = 2\pi/2n$
n = order of rotation axis
Then: $OP \perp PC$
$CA \perp CP$
(Similarly for P' in place of P)

Subroutine CHRTBL

This subroutine prints the complete character table of the molecular point group determined by the subroutine PNTGR.

In addition the subroutine performs a lot of "invisible bookkeeping", namely it sets up all arrays and indices for later use of the character table.

The handling of this character table required by the program is a rather involved and lengthy procedure. Basically the character table is stored in the subroutine and used by the program in a "compressed" form similar to the printed form, i.e. similar to the form usually found in textbooks1). A subsequent program module, which will be discussed briefly, is then capable of "reading" each symmetry class symbol and setting up the symmetry transformation matrices for all the symmetry operations of this symmetry class. With the transformation matrices for the entire point group it is then conveniently possible to carry out any desired further symmetry analysis of a molecule, such as to determine sets of equivalent atoms, internal coordinates, atomic orbitals and to set up symmetry coordinates or symmetry orbitals.

Other subroutines included in the program module " SYMMETRY"

TEST PNTGR: The main program which handles input and output and calls PNTGR and CHRTBL.

MOMIN : Determines the principal moments of inertia.
HDIAG : Diagonalizes a symmetric matrix using the Jacobi procedure. Needed by MOMIN.
ORDMOM: Determines, whether a molecule is a asymmetric, symmetric or spherical top, whether it is linear or planar or nonplanar.
ROTATE: Rotates a molecule by application of the transformation matrix.
ROTAX : Rotates a molecule around one of the coordinate axes.
ROTPNT: Calculates the Cartesian coordinates of the 2 points which define possible axes of rotation for a nonoriented spherical top molecule.
ROTMAT: Determines the desired transformation matrix, given the point(s) determined by PNTROT and the possible order n of the symmetry axis.
TEST : Tests, whether a transformation matrix corresponds to a symmetry operation for the molecule under consideration.
PLTCRT: Plots the molecule. For this a connectivity table of the molecule must be entered.
DATE : Writes the date of the day.

Acknowledgement

The financial support of the Rutgers Center for Computer and Information Services is gratefully acknowledged.

References:

1.a) D. Schonland, "Molecular Symmetry", D.Van Nostrand Company Ltd., London, New York, 1965. b) L.H. Hall, "Group Theory and Symmetry in Chemistry", McGraw-Hill Book Company, New York, 1969. c) J.R. Ferraro and J.S. Ziomek, "Introductory Group Theory: And Its Applications to Molecular Structure", Plenum Press, New York, 1969. d) F.A. Cotton, "Chemical Applications of Group Theory", 2nd ed., Wiley-Interscience, New York, 1971.

2. E.g.K. Mislow, "Introduction to Stereochemistry", W. A. Benjamin, Inc., New York, 1965.

3. E.g. E.B. Wilson, Jr., J.C. Decius and P.C. Cross, "Molecular Vibrations", McGraw-Hill, New York, 1955.

4. E.g. A. Streitwieser, Jr.,'Molecular Orbital Theory for Organic Chemists", Wiley and Sons, Inc., New York, 1961.

5. H. L. Schläfer and G. Gliemann, "Basic Principles of Ligand Field Theory ", Wiley-Interscience, New York, 1969, and references cited therein.

6. S. Bhagavantam, "Crystal Symmetry and Physical Properties", Academic Press, New York, 1966.

7. Ref. 1b), page 67ff. M. M. Orchin and H. H. Jaffé, J. Chem. Ed., 47, 372 (1970).

8. "Report on Notation For the Spectra of Polyatomic Molecules", J. Chem. Phys., 23, 1997 (1955).

The Computer and Its Role in the Development of the Undergraduate Chemistry Curriculum

Joseph R. Denk
Curriculum Development Manager
North Carolina Educational Computing Service
P. O. Box 12175
Research Triangle Park, North Carolina 27709

Undergraduate chemistry provides a strange spectrum of computer activity when one considers that our quantitative science surely should provide the fitting environment to at least the problem-solving mode afforded by a computer. The spectrum of curriculum involvement runs from no usage in large universities as well as small colleges to almost total permeation as at the University of Texas in Austin and at St. Andrews Presbyterian College in Laurinburg, North Carolina. What is strange is that the spectrum does not resolve itself simply into the problem of the "have's" and the "have not's" with respect to availability. Some Universities with graduate schools loaded with computer power haven't a single undergraduate course in chemistry involving the computer. Several small colleges with a single teletype or with a programmable calculator have curricula for which the computer is essential. What is not strange is the fact that the computer is still an infant in education and curriculum development. Even more clear are the financial barriers to usage. Strange and also disturbing are two opposing factors: the departmental non-financial barriers to usage of the computer in education and the apparently forced interest in getting grant money for computer curriculum development as a result of cut-backs in research funds (NSF reports 200 proposals a month for computer-based curriculum development in chemistry).

This session of the Conference on Computers in Chemical Education is dedicated to exploring the frontiers upon which decision-makers embark when they feel they are willing to introduce the computer into their chemistry curriculum. The strange spectrum of usage indicates that there are many on these frontiers and also that many are searching for the frontiers themselves. Further, the doubts as to the validity of the use of the tool relative to its cost seem very probable as contributing to the current state of the art. It is also probable that the modes of potential computer usage still remain unknown to many.

I see the major frontiers to computer-based chemistry curriculum development as being included in one or more of the following five general questions:

(1) What size computer will best serve my curriculum needs?
(2) Should we teach computing itself in the chemistry curriculum?
(3) What computer languages are best for chemistry education?
(4) Should students write all of the programs or is the usage of "canned" programs valid for curriculum development?
(5) What roles can the computer play in the curriculum?

Answers to the last four questions are being pursued in several efforts both with large and small computers. The teaching of computing and the testing of languages are going on in both environments. Programming by students and the use of materials created elsewhere are producing a healthy conflict. In the process, curriculum innovation is emerging as a result of roles played by the computer which are undreamed of by many.

A major issue has emerged on these frontiers -- the issue of whether or not curriculum development rests only on the production of indigenous materials. For those teaching computing itself, doing one's own thing in programming is usually the only way. For the vast majority, however, the hopes lie in the possibility of reaping the programming results of others. Several centers for exchange of computer-based chemical materials are extant for this majority.

EIN was set up to test whether or not programs would be run by remote users on the computer where they exist with I/O going through the mail. This process has proven ineffective. ARPA has attempted to tie-up, by hard-wire connections, several computers so that access would be "direct". A successful hard-wire graphics network exists out of Santa Barbara. The exchange of materials via hard-wire connection is coming but is too far in the future and too expensive for the majority who need materials now. A third mode of exchange, the physical transportation of materials, is occuring at Eastern Michigan University (CECCP)[1], at the North Carolina Educational Computing Service (PALS)[2], at Illinois Institute of Technology (COPES)[3], among several such centers. This mode of exchange is fraught with difficulties unimaginable to the novice but light is being shed and exchange is going on. EIN reported that it catalyzed several hundred exchange efforts even though it was not set up to do this.

The papers selected for this section represent depth in all of the frontiers mentioned above. The first invited speaker, K. Jeffrey Johnson of the University of Pittsburgh, will report on almost all of the frontiers involved in the use of a large computer: teaching programming, creating materials, computer languages, hard-wire exchange, and the sharing of materials. For the choice of computer size, Johnson's presentation will be contrasted with that of Professor Klopfenstein of the University of Oregon who will present the case for the small computer. Professors James and Koch of the University of Northern Colorado will offer contrasting results on the need for teaching computing with those of Professors Klopfenstein and Johnson. These three papers will also offer conflict on the decision for choosing programming languages.

---

[1] Center for the Exchange of Chemistry Computer Programs. For information write: Prof. Ronald Collins, Eastern Michigan University, Ypsilanti, Michigan 48197.

[2] Program and Literature Service. For information write the author of this paper.

[3] Cooperative Program Exchange Service. For information write: Ronald Stiff, Illinois Institute of Technology, Chicago, Illinois 60616.

All of the papers will substantively touch the materials involved in the chemistry curriculum. Al Matsen of the University of Texas, who will present the second invited paper, will bring out some of the most innovative roles of the computer in chemistry education. H. Warren Smith of Earlham College will add further innovation in experimental design. The problems of transportability will become evident in all of these papers and will be touched directly by Larry Sherman of North Carolina A & T State University and K. Jeffrey Johnson. Contrast in opinion on whether or not to use "canned" programs will be evident. A spectrum of materials will be presented running from those which are totally non-transportable to simply movable systems.

It is hoped that these papers help the decision-maker who wants to upgrade his curriculum. To the attendee, this help should come in choosing computer size and programming languages. A key question should enter the attendees mind throughout: Can I obtain materials to get started or should we go from scratch?

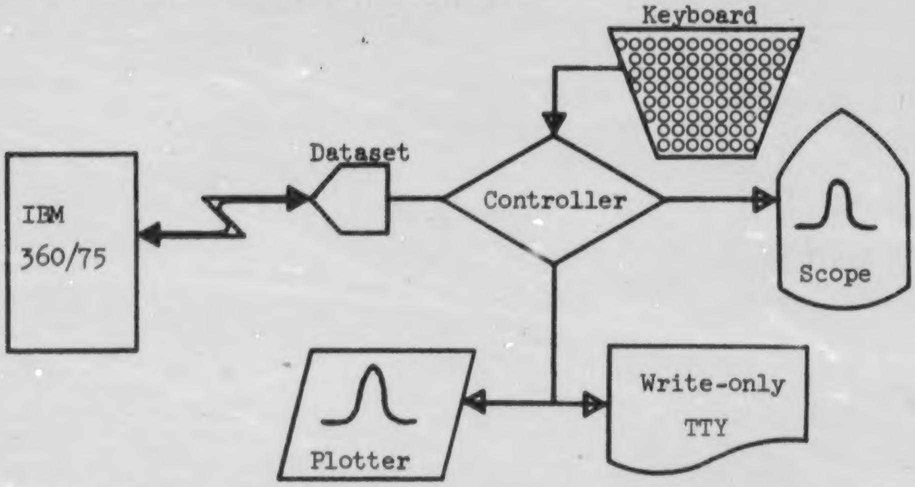
Curriculum Development Through Computer-Based Materials- Graphics, Tutorials and Programming Languages

K. Jeffrey Johnson
Department of Chemistry
University of Pittsburgh
Pittsburgh, Penna. 15213

Computing techniques are being used in a number of ways to increase the effectiveness of chemical education. In this paper I will discuss the results of three applications: an interactive graphics system, tutorial-drill CAI programs and teaching computer programming.

Graphics

Pitt was one of the Universities in the NSF-sponsored interactive graphics network centered at the University of California, Santa Barbara. The software is an extension of the Culler-Fried system, and is supported by an IBM 360/75. A block diagram of the remote terminal is shown below.



The user interacts with the system by pressing keys on the keyboard. Each time a key is pressed a bit string is transmitted to Santa Barbara, an operation is performed, and the result returned and displayed on the scopy. If a permanent copy of the program, numeric values or a plot is desired, the output can be directed to the teletype and/or plotter.

A number of programs to simulate chemical systems have been written. Table I contains a brief description of these programs.

Table I. Graphics Programs

Name	Description
NMR	Simulate AB, AB_2, ABX and A_2X_2, high-resolution NMR spectra. Parameters: chemical shifts, coupling constants.
EXCHANGE	Two site NMR exchange. Parameters: lifetime, shift.
CONTOUR	Constant $\Psi^*\Psi$ contour lines drawn for the 1s,2p,3p and 3d orbitals. Parameters: effective nuclear change, $\Psi^*\Psi/\Psi^*\Psi_{max}$.
RADIAL	Hydrogen atom radial distribution functions (1s,2s,2p 3p,3d). Parameters: quantum numbers.
EQUIL	Calculate equilibrium constant by minimizing total free energy function. Output: total free energy vs. extent of reaction. Parameters: temperature, stoichiometry, standard chemical potentials.
HEAT	Enthalpy of reaction. Integrates empirical Cp function. Output: display of function, integral. Parameters: empirical constants, temperature range, enthalpy at one temperature.
KIN (1)	Simulates $A \xrightarrow{K1} B \xrightarrow{K2} C$. Output: A,B,C vs. time. Parameters K1, Ka, A_o.
KIN (2)	$A \rightleftarrows B \rightleftarrows C$. Parameters: 4 rate constants, A_o.
DERIV	Determines the inflection point of a sigmoid curve.
EDTA	Simulate M^{2+} - EDTA titration. Parameters: K_f, pH, $[M^{2+}]$.
NERNST	Calculate EMF for $M\|M^{z+}$ electrode in H_2O-NH_3 solution. Output: EMF vs. pH. Parameters: Z, Eo, successive formation constants

Table I. Graphics Programs (Continued)

Name	Description
MX_2	Calculate solubility curve of MX_2, where X is conjugate base of HX. Output: solubility and log of solubility vs. pH. Parameters: Ksp, Ka.
LIGAND	Metal-ligand, equilibria. Output: fraction of each species vs. log of ligand conc. Parameters: number of ligands, stepwise formation constants.
$\overline{N}$	Simulates formation function in metal-ligand systems. Parameters: number of ligands, stepwise formation constants.

The output of these simulation programs has been recorded on Super-8mm film, and 35-mm slides and prints. Film loops are currently being prepared and the slides and prints can be used in several courses.

During the fall term of 1970, a memo was sent to all students enrolled in physical chemistry courses. Interested students were invited to sign up for special one credit course in programming the interactive graphics system. The course met one evening each week. Eight students enrolled. After two hours the students were able to start programming. The only requirement was a documented project due at the end of the term. Most of the students chose their own projects. They included: a simulation of van der Walls forces, a study of the solubility behavior of silver halides as a function of halide concentration, an analysis of the Michaelis-Menten equation and Lineweaver-Burk plots of enzyme inhibition systems, and a simulation of the Monod-Wyman-Changeux model of enzyme cooperativity. One of the students was so hooked on the system that he wanted more information about the NSF proposal. He was going to graduate school and wanted to get a terminal.

Tutorials

The Pitt Computer Center supports the CAI-oriented language CATALYST/PIL. CATALYST (Computer Assisted Teaching and Learning System) is a processor designed to facilitate the writing of CAI lessons. PIL is a JOSS-derived conversational language. The CATALYST/PIL interface provided both lesson designer and student easy access to desireable features of both languages. The student, for example, can use the desk calculator mode of PIL to solve chemical arithmetic problems. The lesson designer can build in random numbers wherever desireable so that no two students receive the same problems. It is also easy for the lesson designer to build in branches if the student asks for help, more information, is off by a factor of two, etc.

The emphasis has been on numerical problems. There are 20 tutorial-drill lessons on stoichiometry, gas laws, colligative properties, Faraday's laws and aqueous equilibria. There are 4 verbal lessons on bonding theories, periodic properties and electrolytes. The lessons average 20 minutes of terminal time.

In principle this approach has a number of advantages. The computer is an infinitely patient tutor. Students can use PIL to solve stoichiometry and pH problems. At any point the student can type help and get an appropriate hint. If the student types a wrong answer, the computer will analyze it for the two or three most common mistakes and write an appropriate message ("Did you forget to convert from degrees Centigrade to degrees Kelvin?"). After two or three incorrect responses the computer will outline the correct solution to the problem.

There are also a number of disadvantages. A time-sharing system is required. Free time and an adequate supply of terminals must be available. The system must be reliable, and easy to use. The lessons require a tremendous investment in programmer and lesson-designer time.

The students seem to enjoy using the programs when the time-sharing system is stable and they can find a terminal. They ask me when programs on the Bohr atom, radioactivity, hybridization, etc. will be available. I think that with care an effective library of tutorial-drill modules can be developed that will make freshman chemistry a more stimulating learning experience.

Programming Languages

An attempt is being made to make computing an integral

Analytical Chemistry (Continued)

Name	Description
TITR	Data reduction program, determines thermodynamic acid dissociation constants of β-alanine from experimental data.
TITRN#	Data reduction program, numerical differentiation of titration data.

Physical Chemistry

GAS	Calculate compressibility of a van der Waals gas as a function of P, a, b, and T.
EQUIL*	Calculate equilibrium constant by minimizing total free energy function, function of stoichiometric factors, chemical potentials and temperature.
HEAT	Calculate enthalpy of reaction from $\Delta H_{T_2} = \Delta H_{T_1} + \int_{T_1}^{T_2} \Delta CpdT$
BOX	Solve particle in the box problem: $\tan \left\{ \frac{2a(2mE)^{1/2}}{\hbar} \right\} = \frac{2(Vo-E)^{1/2}E^{1/2}}{2E-Vo}$ function of a, and Vo.
EMF	Perform regression analysis on EMF data as a function of molality.
VOLUME	Determine partial molal volume from solution data.
ENTROPY*	Calculates the entropy of a molecule from spectroscopic data; function of the number of atoms in the molecule, linearity, temperature, pressure, symmetry factor, multiplicity, fundamental vibration frequencies, mass of each atom and the coordinates of each atom relative to an arbitrary origin.
NMR*	Calculates an AB, AB_2, ABX, A_2X_2 or A_2B_3 NMR spectrum as a function of chemical shifts and coupling constants.

part of undergraduate instruction in chemistry at Pitt. The freshman use PIL in a desk calculator mode as described above. In analytical chemistry an optional problem set is available in which certain chemical systems are to be simulated. A library of Fortran programs is also described so that students can check their work and use the computer to explore these systems as a function of the parameters. Another problem set is available to physical chemistry students. In addition, two hours of instruction in PIL are included as part of the laboratory-recitation program. This is sufficient background for the students to write their own linear least squares program.

An upper-division elective, "Numerical Methods in Chemistry", is given once a year. In this course the PIL and Fortran IV languages are covered and several topics in numerical analysis are discussed from a chemical applications point of view. This course is described in J. Chem. Ed. 47, 819 (1970).

The current library of Fortran programs is described in Table II.

Table II. Fortran Programs

Analytical Chemistry

Name	Description
HA*	Simulate weak acid-strong base titration curve; function of Ka, Ca.
EDTA*	Simulate EDTA-M^{2+} titration curve; function of Kf and pH.
NERNST*	Calculate EMF for $M\|M^{Z+}$ electrode in M^{Z+} -NH_3 solution; function of pH, Ki, Z, E°.
MX2	Calculate solubility of MX_2 where X is conjugate base of HX; function of Ksp, Ka, pH.
MX*	Calculate solubility of MX where X is the conjugate base of H_zX; function of Ksp, K_1, K_2, ... K_z, pH.
MA#	Calculate solubility of MA where M forms complexes with a ligand L; function of Ksp, K_1, K_2, ... Kn, log [L].

Physical Chemistry (Continued)

Name	Description
ABC*	Calculates an ABC NMR spectrum as a function of chemical shifts and coupling constants.
HMO*	Solves Hückel Molecular Orbital problems.
CONTOUR	Calculates orbital contours; function of Z_{eff} and $\Psi^*\Psi$.
LINEAR*	Data reduction program, performs linear regression analysis.
GUG	Data reduction program, LINEAR modified for the Guggenheim method.

Others

IR*	Searches a database of organic compounds containing melting and boiling points, strong IR band frequencies and Sadtler file members.
POLREG*	Polynomial regression analysis.
MATINV*	Matrix inverter.
JACOBI	Hermetian matrix diagonalization routine.
RUNGE	Solves a system of differential equations.
SECANT	Solves a system of nonlinear simultaneous equations (Secant method).
NEWTON#	Solves a system of nonlinear simultaneous equations (Newton-Rapheson method).
ITINV#	Interactive matrix inverse routine.
NONLIN #	Nonlinear least squares routine.
PLOT*	Typewriter art plot routine.
CALCMP #	Calcomp plotter routine.

---

*Complete documentation available. #In preparation.

Concluding Remarks

I have emphasized here two applications of computers in chemical education. The one, through CAI, is a tutorial-drill facility. And the second is a facility for simulation and problem solving. CAI may be able to provide poorly motivated freshman chemistry students an interesting and effective teaching tool. The simulation programs will allow good students to manipulate chemical systems as a function of the parameters. These programs can provide a relatively painless procedure for studying numerically complex systems. Carefully chosen computer-based curriculum materials can add a new dimension to chemical education.

Acknowledgements

Support of this work from the National Science Foundation Grants GJ-696 (graphics) and GU-3184 (institutional science development) is gratefully acknowledged. I particularly wish to thank the many programmers, undergraduate and graduate students who have contributed to the project, particularly Ellene Tratras (CAI), Dennis Doerfler (Fortran) and Susanna Levitt (graphics).

APPLICATION OF "CANNED" PROGRAMS AND COMPUTERIZED LITERATURE SEARCHING TO AN UNDERGRADUATE CHEMICAL CURRICULUM[1]

By
Larry R. Sherman

Department of Chemistry
North Carolina Agricultural and Technical State University
Greensboro, North Carolina 27411

INTRODUCTION

The use of computer technology in the chemical curriculum at A & T State University can be divided into three parts. (1) The conventional use of programming in the instrumental analysis course to aid in calculating the data derived from complicated experiments; (2) The use of canned programs in the normal chemical curriculum; (3) The development of a course in computer literature searching and the retrieving of data from a data bank.

There are two computer facilities available to faculty and students at A & T State. There is a Computer Science Department on the campus which operates a Control Data 3300 computer (CDC). The computer will accept Fortran IV, Cobol, and extended Algol programs and is adequate for teaching computer programming, administrative operation, and processing research data; however, all data is batch processed and the computer facility lacks the versatility desired for classroom instruction. It has the advantage of twenty minute to one hour turnover time during the seventy-seven hours it operates per week. The second facility available is an IBM 1050 terminal connected to the Triangle University Computer Center (TUCC) at the Research Triangle Park. The terminal is located in the Engineering Building. TUCC has an IBM 360 Model 75 computer which is exclusively devoted to time sharing; this computer will accept many more languages than the CDC, including PL/1, WatFour, WatFive, and conversational program (cps). Since almost one hundred terminals throughout the state of North Carolina feed data in the computer, the turnover time is often five to seven hours during busy times in the Universities schedules. However, the disadvantage of slow turnover time is compensated by the vast data bank which is available; and since a large number of programs are available in conversational programming, direct classroom instruction can be given through the terminal. Due to the differences in the systems, direct translation from one system to the other is not possible and minor changes in programming are necessary, beside changes in job card and system cards, even when the same language is being used. Thus elementary student programs become limited to one system.

PART I

Because of the lack of transportability all programming instruction in the chemistry department was adapted to the CDC. Fortran IV programming is introduced into the Instrumental Analysis class. The students receive four lectures on programming, giving them sufficient background to use the computer to perform linear regression analysis and give statistical interpretation of their experimental data. Although the students do not learn any sophisticated programming, they have learned the fundamentals and application of data processing to non-trivial experiments.

PART II

Other than the instrumental analysis course, the Chemistry Department does not offer its students formal programming instructions; the students are expected to obtain the instruction in other departments. In other chemistry courses, the computer is used as a "black box." It is often difficult to convince people of the merits of computer black box instruction. Some sophisticated programmers feel the black box usage has little educational value. Some novices feel the computer is a gimmick. Although both arguments have merit, the people who put forth these arguments fail to look at the purposes behind the computer usage. They see the computer as a tool in itself and not as a means to teach other subjects. The computer should be viewed in the same manner as the infrared (IR) or nuclear magnetic spectrometer resource (NMR). Every chemist is expected to know the rudiments of interpreting infrared spectra, but rarely understand the internal workings of the instrument or the quantum mechanical reasons for the infrared bands. Likewise most chemists can give an intelligent discussion of magnetic shifts of an NMR but few understand the difference between homogeneous and nonhomogeneous fields and fewer chemists can solve the exponential Hamiltonians which explain the observed shifts. Like the IR and the NMR, a computer can be used to teach chemical principles without an understanding of the theory behind its use. Just as full use of an IR or NMR demands a spectroscopist, the full use of a computer demands a skilled programmer. However, other users only need to know what kind of data they desire, and which buttons to push.

QUIZ

Some schools, like the University of Pittsburgh, have developed computerized instructional homework but the faculties and time necessary for this type of programmed learning are generally prohibitive. The program QUIZ[2] is a computerized set ot typical chemistry problems. It contains general chemistry and quantitative analysis problems that are worked by the student as homework. The student works the problem like any homework assignment, then he puts his answers on a punch card along with his name and student number; the punch card is delivered to the instructor who places it in the program deck. The computer checks the student's answers and if an answer is correct it identifies the question and prints a congratulatory sentence. If the answer is incorrect, the computer prints a warning that it will only accept answers with three significant figures, then prints the correct format for solving the problem and asks the student to rework the question. The computer also keeps a running tally of the number of correct answers submitted by each student and prints a grade on each set of questions. Although this grade could be used for computing students' grades, it is not recorded because a keypunch error can not be distinguished from an error in chemical theory nor is it possible to distinguish resubmitted problems from those assigned for that week.

With QUIZ the student's work is corrected the same day it is submitted and he received tutorial help with any problem which causes him difficulty. QUIZ also allows the student to resubmit incorrectly worked problems and have them graded; this would not normally be practical if questions were graded by the instructor. Complicated redox equations are included; the student enters the sum of the coefficients for reactants and products. Since only redox equations are used, the probability of the sum being correct when the equation is balanced incorrectly is minimized.

QUIZ is written in Fortran IV and can hold 999 typical chemical problems. The student is given a mimeographed sheet containing the problems. Some of the problems are complete, other problems have data missing; the missing data is supplied by the student, using the least three digits of his student number. The use of blanks gives some individuality to each student's work and decreases the copying of answers. The student must use his own student number on the answer card; the computer picks up the last three digits of the student number and inserts it into the problems which is stored in the program. The final printout is returned to the student by his number.

Because of the additional time needed by the student to prepare a punch card, the response to computerized homework has been less than with conventional methods.

HA

HA[3], a program for preparing an acid-base titration curve, is one of the most useful programs in teaching analytical chemistry. The program was first introduced into the Quantitative Analysis course. The student enters the normality of the acid, normality of the base, number of ml of acid and pKa for the acid onto a computer card, and the computer returns twenty-four points for pH vs % titrated curve from 0% to 120% of base. The student then plots the data on a piece of graph paper to see the effect of a change in pKa on a titration curve. Since the computer time is less than two seconds per run, it is possible to have each student prepare a titration curve for five pKa values from one to nine. Normally, a student could not be expected to plot a curve for more then one pKa value in an assignment because of the tedious calculations.

Currently the program solves a quadratic equation but it is being extended to the solution of cubic equations so that pKa values of greater than nine can be used without changing the basic format. This program is mounted both at TUCC and on the local CDC system, so that only a minimum of student time is required in obtaining data.

POLTRA

Another program which has considerable use is the POLTRA program[4]. It is a program for solving polynominal equations and has a great deal of use in helping to avoid the tedious calculations involved in computing the pH of a weak acid or base when the weak electrolyte's concentration is such that a significant portion of the electrolyte has dissociated (10% or greater). In a complex pH problem, five to twelve non-linear equations are obtained with the same number of unknowns as equations. These can be reduced to one polynomial equation which is then solved by numerical analysis; the latter may involve hours of hand calculations especially if the initial guess is poor. In the POLTRA program, the student enters the order of the polynomial, and the coefficients in ascending order. With these the computer calculates all real and imaginary solutions to the problem. The student can then use the positive real numbers to determine the exact pH of the solution. This program has made it possible to assign a large number of problems which illustrate the change in pH with a change in concentration, pKa, or pKb without requiring the student to perform the lengthy arithematic. It is also possible to illustrate the leveling affect of water on the pH of the solution. Often the latter is neglected because the solution of the equations is "too difficult," and the instructor allows the students to make invalid assumptions because the answere can not easily be calculated.

POLTRA is a good example of using the computer to perform calculations so that detailed chemical principles can be studied.

ISIS

ISIS[5] is a program for searching the American Society for Testing Materials (ASTM) infrared file mounted at TUCC. ASTM has gathered all the major IR files (Sadler, AIPE, AMC, etc.) and has computerized the searching of 92,000 IR spectra. This program has been mounted at TUCC and can be reached through remote terminals. The program is most efficiently run if the no-band region of spectrum is first determined; a no-band is a portion of the spectrum where no vibrations produce a band of more than 30% transmission. Next the wave length of the IR bands are determined and the computer searches the file to identify the spectrum. To compensate for differences in instruments, a "wiggle" factor of 0.1 microns is provided for each peak wave length. The computer prints out the code numbers of all the compounds with spectrum, which corresponds to the entered data. Usually three to ten code numbers are obtained, but since more than one file may contain the same compound, the "hit" ratio is much higher than would iritially appear from the printout.

In the instrumental analysis course, the students obtain the infrared spectrum of a solid and liquid unknown (usually a simple organic compound like DMF, DMSO, -napthanol. They first try to identify the compound using reference books. After a week's work - usually unsuccessful - they are instructed in the use of ISIS and receive the answer to their unknown in about twenty minutes.

ISIS has also been used by the students taking Qualitative Organic Analysis. Although ISIS was not introduced into the course, the students quickly learned to use it, and began identigying all their unknown by IR. Unfortunately, the instructor who taught the course did not recognize the potential of computer searching and felt the students had just learned to "beat" the course.

GAME

GAME is a qualitative organic unknown game. It is proprogramed with organic compounds and forty qualitative analysis tests for these compounds. Each test is assigned a cost which is based on the time it would take an analyst to perform the test. The object of the game is to determine the unknown at the lowest cost. Since an understanding of the tests is essential to identify the compound, a great deal of organic chemistry can be learned with GAME.

Faculty members usually enjoy playing the fame; however, students sometimes become frustrated because they may run fifteen to twenty tests and still have no answer to their unknown. GAME has limited usefulness in the North Carolina Educational Computing Service (NCECE) system, for the students quickly learn they can obtain the IR for their unknown. They punch the data on a computer card and run the ISIS program to identify the unknown. When this happens, GAME becomes useless and it is put away until the next class.

If one of the objects of a modern chemical curriculum is to teach students to interact parameters, GAME and ISIS have done this. Computer technology does not only save time and open new avenues of study, it also frustrates instructors whose classical teaching methods become obsolete. It, more than anything else, demands that new approaches and ideas be introduced into chemical curriculum so that the normal curriculum does not become trivial.

PART III

At A & T there is a catchall course entitled, "Current Trends in Chemistry." In the summer of 1970 it was decided to use this course to teach computerized literature searching and data retrieval.

The course is a sophomore chemistry course, all students having completed at least general chemistry, quantitative analysis, and organic chemistry. The majority of the students have not had a course in computer programming and this course is taught with the same black-box concept described earlier. The students are given a very elementary introduction to computer usage and encouraged to ask questions about programming, but no programming per se is taught. The students are instructed in keypunch operation for batch processing of data and the use of a teletype for CPS work; the latter is used whenever possible. About one third of the course is devoted to literature search using an unconventional method. One third is devoted to data retrieval and processing and the remainder to studies of employment and problems in the chemical industry. Searches are not conducted to be complete but to be rapid. This means a great deal of literature might be overlooked, but in light that the majority of our students obtain industrial positions, where useable data on a subject is more important than complete retrieval, speed is emphasized over total recall. At first it was thought that the Computer Readable Service of Chemical Abstracts (CA) would be ideal since it is the largest data bank available. Burlington Industries, the only local subscriber to this service, offered their facilities. However, they discontinued subscription to the CA tapes before the class met. Contact was then made with Mr. MacDonald, the Marketing Manager at Chemical Abstracts, who offered to sell a sample package of tapes, programs, etc. for $40. The package could be mounted on any computer system and be repeatedly used, but because of a change over at CA to a standard distribution format, the package would have been obsolete before it could be used a second time in the course.

Since the CA file is the largest and most complete scientific data abstracting and editing service, it could not be neglected. The students were introduced to the purpose and historical background of CA's computer readable service. They were given the cost of buying the service and the alternate of subscribing through a data processing center. Typical "file" cards were distributed to the students and an explanation given of all the information on the cards, with emphasis put on the indexing search words.

An alternate to the CA search was provided by the Science and Technology Research Center (STRC) at the Research Triangle Park. They conduct searches on the NASA, DOD, TI and EI files on a contract basis for universities and industries. They have also prepared a demonstration file on the Textile Industry (TEX2)[7]. This file is stored at TUCC and assessable through the remote terminals. Permission was given by STRC for students and instructor of this course to use the file. The searches are limited because of the amount of data in the film, but illustrate many important points.

The TEX2 file is designed to search two subjects at once. While the file is being searched for a subject, another subject can be added to the search and all references to both subjects can be recalled from the system; or while a subject is being searched it can be intersected by another subject and all references common to both subjects can be recalled from the system. Information which is unrelated to the subjects desired can be programmed to be deleted. For example, a search for references on formaldehyde resins would produce references for both phenolic and acetone resins. If only references for phenolic resins are desired, references to acetone resins can be deleted by instructing the computer to delete them from the list. However, of the three search techniques, addition, intersection and deletion, the latter is the least useful, because it is often better to retrieve extraneous references and manually eliminate them than to allow the computer to eliminate references which might be useful. However, because of the size of the TEX2 film it has not been programmed to delete.

An index of key words and the number of times they are used has been prepared to show the total number of paper references under any subject matter. This gives the searcher an indication of the fruitfulness of his search. If a subject has only three or four references, an intersection or deletion could eliminate all the information on the subject, giving no base for the continuation of a search. Large subject items (500 to 20,000) would produce so many estraneous references without intersection or deletion that a computer search would have no advantage over a manual search.

The TeX2 file which is written in CPS has a limited number of references and keyword and is programmed to print out the nearest word to a misspelled or nonexistent keywords. When a keyword is requested, the computer prints out the number of references and asks how many are desired. This file, which was prepared by textile students, also demonstrates the problems which can arise when technicians in one field prepare an index to be used in all fields. There are sixteen references filed under the keyword Heat and two references under Entropy, a term usually associated with heat. If references for Heat were requested and intersected with a request for entropy references, two printouts would be expected. However, the recall was zero. The textile students who prepared the file had studied little or no physical chemistry and did not recognize that Entropy should be a subdivision of Heat. The failure of this search illustrates that even computer searches are strongly dependent upon the technical knowledge of the programmer and indexer.

The TEX2 file was demonstrated in one laboratory-lecture session. The students were asked to prepare a literature search based on the file and execute it through the terminal. They were given an outline and an index and asked to make additions to the main subject, intersections and deletions (the latter to familiarize the students with the technique). From their work, they prepared a report on the use of the file. While the students were preparing a literature search on the TEX2 file, a real literature search on electrochemistry was also prepared by the instructor using the NASA file as STRC. After the searches were completed, the entire class made a visit to STRC where they were instructed further in literature searches and given demonstration of the working of a search institution. This gave the students the insight into the vast amount of work and expense put into setting up and operating an organization which can handle the literature published in the scientific fields. At the time of the visit Mr. Lockwood, the

Applications Engineer, illustrated by using the instructor's prepared search, the tedious process of deleting, after retrieval, the extraneous information which naturally comes with the search due to non-specificity of the English language. He illustrated this by explaining that during a search for references on Venetian blinds, the computer will also retrieve references on blind Venetians and any attempt to delete the latter runs the risk of invalidating the search.

ACKNOWLEDGMENTS

The author wishes to thank the North Carolina Educational Computing Service for an award which helped to pay for the computer usuage and also to thank Mr. Peter Chanery and the staff of the Science and Technology Research Center for their assistance in this work.

NOTES

1. All programs mentioned in this talk, except the TEX2 file, are available from the author or from Dr. Joseph Denk, N.C.E.C.S., Research Triangle Park, North Carolina.
2. Prepared by L. R. Sherman, North Carolina A & T State University, Greensboro, North Carolina.
3. Prepared by J. Johnson, University of Pittsburgh, Pittsburgh, Pennsylvania.
4. Preparge by S. Painter, N.C.E.C.S., Research Triangle Park, North Carolina.
5. Prepared by J. Denk, N.C.E.C.S, Research Triangle Park, North Carolina.
6. Prepared by F. Hornak, University of North Carolina, Wilmington, North Carolina.
7. The programming for the file was written by M. A. Williamson, S.T.R.C., Research Triangle Park, North Carolina. The indexing for the file was prepared by students at North Carolina State University, Raleigh, North Carolina, under the direction of Dr. Robert Work.

A COMPUTER-BASED SIMULATION GAME
FOR TEACHING EXPERIMENTAL STRATEGIES IN CHEMICAL KINETICS

H. Warren Smith
Earlham College
Richmond, Indiana

Laboratory exercises have traditionally been an important part of science education and especially chemical education. They have served to get our students past the world of words, words in books or words in lectures, to the real world of materials. In the laboratory the student observes how chemicals behave and begins to see that the chemical concepts presented in lecture are useful in understanding chemical phenomenon. In the laboratory the student is introduced to a variety of chemical techniques and gains confidence in his ability to manipulate certain aspects of nature.

In addition to introducing the student to the ritualistic aspects of chemistry, the laboratory can also be used to develop other skills used by the research scientist. As the student gains confidence and ability in the experimental techniques, he is encouraged to take a larger role in planning the experiment. Eventually the laboratory may become open-ended to the extent that the student is given a statement of the problem to be investigated and asked to completely design the procedure, carry out the experiment and interpret the data. It is these two additional factors -- experimental design and analysis of the data -- that we believe should be emphasized in the laboratory training of our students.

Computer Roles in Chemical Education

The computer has been used in several ways to improve laboratory instruction. Most of the student use of the computer is essentially as an extension of his slide rule. He learns to use it to avoid the laborious computations needed to reduce his experimental data to results that can be understood in terms of the concepts of chemistry. For example, the results of a number of titrations are averaged and the concentration of the solution calculated by the computer. I would expect that the biggest use of computers by chemists has been in this data processing mode.

Computers are also used, in both teaching and research, to simulate chemical systems with a computer-based mathematical model. Complex reaction mechanisms can be readily simulated by a series of differential equations, so that the concentrations of any of the chemical species in a reacting system can be printed or graphed as a function of time. These calculated curves can be compared with experimental data as a test of the validity of the model. Pedagogically this approach can be used to help students discover relevant parameters of a system and the functional relation of these parameters. For example, a computer programed with a model of gas behavior might be used by a student to discover the ideal gas law by playing around with volume, temperature and mass to see what gas pressures will result. A more sophisticated simulation might lead him to discover something about the behavior of real gases. Hence, the simulation can partially replace a laboratory experience. There is also an increase in educational efficiency, since the time-consuming process of data collection can be omitted.

Generating individualized data for "dry" laboratory experiments is a use of the computer closely related to simulation. In this pedagogical technique, an experiment is described to the students--perhaps they are shown a film of an experiment being performed. They are then supplied with their personal set of data, which might have been obtained from this experiment. The student struggles with the problem of data reduction and interpretation in much the same way as in the traditional lab. However, in the case of the computer-generated data, the student is certain that, while the data may contain "experimental" errors, the basic design and execution of the experiment are such that a meaningful analysis is possible. In the traditional lab exercise the experimental design has been worked out for the student, but his failure to follow the instructions carefully may result in uninterpretable data. For example, the student who discards the wrong fraction of a separation may be required to perform the probably unrewarding exercise of repeating the experiment. The use of computer-generated data again produces time saving by eliminating the data collection step.

Another use of the computer, which has not been as widely exploited in chemistry as in economics and business administration, is the use of games.[1] The clear advantage of a game is that it must have explicit goals. The student knows when he is winning. The second major aspect of a game is that it has a set of rules. The rules may be highly artificial, as in football where foreward passes are only allowed from behind the line of scrimmage. In applying the concept of games to chemical education, probably the most useful games would have rules very similar to those used in simulation. For example, a chemistry game based on gas behavior would deal with pressure, temperature, volume and mass in ways that are designed to model what might reasonably happen in the laboratory. However, the rules are under the control of the instructor and may be modified to achieve his educational objectives.

The rules relating to how the game is scored are particularly useful in shaping student behavior. The size of the ante required to get into the game may be made dependent on the amount of information sought from the simulation. The size of the payoff could depend in various ways on the type of results obtained. By adjusting the cost-payoff formulas a winning strategy could be made to correspond to the behavior of a trained chemist.

KNEXP: A Kinetics Experiment Game

To begin the kinetics-experiment game, KNEXP, the player makes a number of choices which determine the design of the experiment. To study a designated reaction, he selects the initial concentrations of reactants, the temperature of the reactants, the product or reactant which is to be sampled, and the number and frequency of samples to be made. The player must also specify the degree of precision desired on each of these variables. The computer then prints a table of the desired concentrations, along with a plot of these concentrations vs. time. The student is then faced with the task of analyzing a set of data which contains "experimental" errors that depend on his specified precisions. The analysis is further complicated by the student's basic experimental design, which may or may not lead to interpretable data. The data may show only that, in the case of a relatively slow reaction, essentially no changes have occurred in the time interval examined, or that, in the case of a relatively fast reaction, the reaction has gone to completion before more than one or two data points can be collected. Less obvious difficulties can arise when conditions for "flooding", or the method of initial rates are not met; or the data points cover only a fraction of a half-life.

Problems related to experimental design are seldom allowed to occur in the usual laboratory exercises. In most cases, design errors are programed out of the students' experience by the laboratory manual author or the course instructor. This is necessary because of the large amount of time and effort that could be wasted by collecting uninterpretable data. Since the student does not have the opportunity to try various approaches to a problem, including some obvious blunders, he does not have the opportunity to learn why some experiments give useful data and others do not.

In order to convert this simulation of a kinetics experiment into a game, a charge is made for each kinetics run, which is inversely related to the degree of precision specified on each variable and directly related to the number of experimental concentrations that are analyzed. When a student reports his result, in this case a rate law and activation energy, he is awarded a certain payoff, which hopefully balances off the charges accumulated in obtaining the result. In any event, he has a score, which tells him whether he is winning or losing.

The assessment of a charge can also be used to improve the simulation. In this case the ante required would be related to the time and effort needed to actually perform the experiment in the laboratory. This is the service the computer is performing for the player. Hence, the charge is graduated to indicate the relative difficulty of preparing and analyzing solutions or of maintaining a selected temperature interval. The charging formula makes it too expensive to specify all variables to the highest attainable precision. The player is forced to consider the trade-off between the rising costs of his experiment and the probability of a useful outcome to the experiment. Students overly impressed with the cost of precision have produced data which was sufficiently random that it fit equally well (or poorly) a zero, first, and second order kinetics plot. On the other extreme, by specifying very high precision or a large number of data points a player may incur a larger expense than any positive payoff he might receive.

The Programing Strategy for KNEXP

An abbreviated flow diagram of the program, KNEXP, is shown in Figure 1. The reaction specifications are read from cards and printed for the student to check. If there has been a card punching error, the run can be halted at no charge. Otherwise, the program selects the stoichiometric coefficients and the Arrhenius temperature-dependence parameters from a stored table. In the present program the reactions and rate laws are chosen to be reasonable, but do not correspond to actual reactions.

The experimental errors are simulated using the IBM-supplied scientific subroutine GAUSS, which selects a random number from a normal distribution with a specified mean and standard deviation. A random initial index for GAUSS is obtained by having the student flip a data switch, which terminates a loop. The exit value of the loop index is then used by GAUSS. In this way it is assured that no two runs will be the same even with identical input data. Using this method, initial values of the concentration of all reactants are chosen using the student's precision specification as the standard deviation. These values are not further randomized during the run. The values of temperature and the time are selected randomly at each time specified. The initial concentration, time and temperature values are used to compute concentrations at a later time. The errors resulting from analysis of the sample are introduced by a final call to GAUSS, which uses the student's specification of precision of analysis as the standard deviation. Hence, the values of temperature, time and sample concentration vary randomly about their "true" values during a kinetic run, producing a spread of data points about the best straight line on the correct kinetic plot. However, a lack of accuracy in the choice of initial concentration will not effect the spread of data points, if they are from a single experimental run.

The computer program uses integrated rate laws to calculate concentrations of reactants and products at the times specified by students. At present three cases have been programed: first order reaction in one reactant, zero order in others; second order in one reactant, zero order in others; and first order in two reactants and zero order in other reactants. Rate constants for these equations are calculated using the Arrhenius form of the temperature dependence equation. Other more complicated rate laws and sophisticated temperature dependence equations could be employed as the student's knowledge of chemical dynamics becomes more advanced.

The program identifies the student by a code number and calculates his ante for the run by a formula such as the following:

$$\text{Cost} = \Sigma \frac{0.05}{\delta([A]_o)/[A]_o} + \frac{0.5}{\delta(T)}$$

$$+ \left(\Sigma \frac{0.05}{\delta([A])/[A]} + \frac{0.5}{\delta(t)}\right) \cdot N$$

where

$[A]_o$ is the initial concentration of reactant A

$\delta([A]_o)$ is the precision of $[A]_o$

$\delta(T)$ is the precision of the temperature ($^\circ$C)

$[A]$ is the concentration of compound A at time t.

$\delta([A])$ is the precision of $[A]$

$\delta(t)$ is the precision of the time (sec)

N is the number of samples analyzed.

The first summation is over all reactants, while the second is over all compounds being analyzed. The constant factors in this equation are chosen to give relatively even weight to the various factors. However, if the instructor wished to encourage the use of the method of initial rates, for example, the constants in the first and second terms could be lowered relative to the constants in the last term. In this way the cost of setting up a number of kinetic runs would be reduced relative to collecting a number of data points with the same initial concentrations. Even without this added incentive, students soon see that four or five data points per kinetic run are usually sufficient, provided the points cover an adequate range of reaction - a fact often missed by students working in the laboratory, perhaps because of the separation of data collection in the lab from the later data analysis at home. Also in the computer game the cost of additional data points is more explicit.

In addition to tabular and graphical output of the data resulting from a run, the data may also be punched on to cards for processing by other programs. In conjunction with KNEXP, a general graphing program is made available to the students to aid their analysis of the data. The graphing program allows the data from the kinetic experiment game to be plotted in the usual zero, first, second or third order kinetics plots over any range of time-concentration values. The slope and intercept of the least squares line through the data points are also calculated. One student, in an attempt to get more information from a run, wrote a computer program that systematically calculated sets of rate constants for various reaction orders. This program made it possible, at least in principle, to obtain the rate law from one or two carefully chosen experiments.

The payoff for the correct rate law and reasonable values for the specific rate constant at 25°C and the activation energy depends on the difficulty of the reaction and the accuracy of the results. Such factors as number of reactants and overall order of the reaction are considered in assessing the difficulty of the problem. Values within ± 20% of the "true" value of the specific rate constant and ± 0.5 Kcal on the activation energy are considered acceptable for full credit.

Conclusion

The type of laboratory exercise represented by this reaction kinetics game shares many of the features of the more open-ended, problem-oriented laboratory experiments. It engages the student's attention in more than careful reading of the procedure. He selects the procedure himself and works out the details. The analysis of the data is not a pro forma exercise to reach a conclusion already obvious from reading the text. On the other hand, it has the advantage over many "research" type experiments that the rules of the game can be closely controlled by the instructor, and the time-consuming process of data collection is omitted. A common problem with open-ended labs is that after spending several weeks designing the experiment, collecting data, and analyzing the results the student realizes that his design was faulty or that other experiments must be done in order to achieve his goal. In the computer game, he would have ready access to more data and other experimental designs. In addition, the instructor can present the problems in order of increasing difficulty, so that a student is challenged, but succeeds in reaching the goal.

In comparing the values of the open-ended laboratory with the game-type of laboratory, the educational efficiency of these pedagogical techniques should be considered. Many more situations in chemical dynamics can be presented to the student using the strategy of the computer game and eliminating the educational unproductive data collection exercise.

The rapid acquisition of data frees the student's time and effort for concentration on the important skills of experimental design and interpretation of experimental data. Finally, there is an intrinsic motivational factor in a game. Since the game is played against the system (nature) rather than against their classmates, it is possible for everyone to win.

This is not to say that all wet laboratory experiments should be replaced by computer-based simulation games, but that some mix which includes games is definitely appropriate.

Footnote

1. The inspiration for the program discussed in this paper came from DATACALL, a simulation game programed by Richard Johnson to teach research strategies in psychology at Earlham College. DATACALL was described at the Dartmouth Conference on Computers in June 1971.

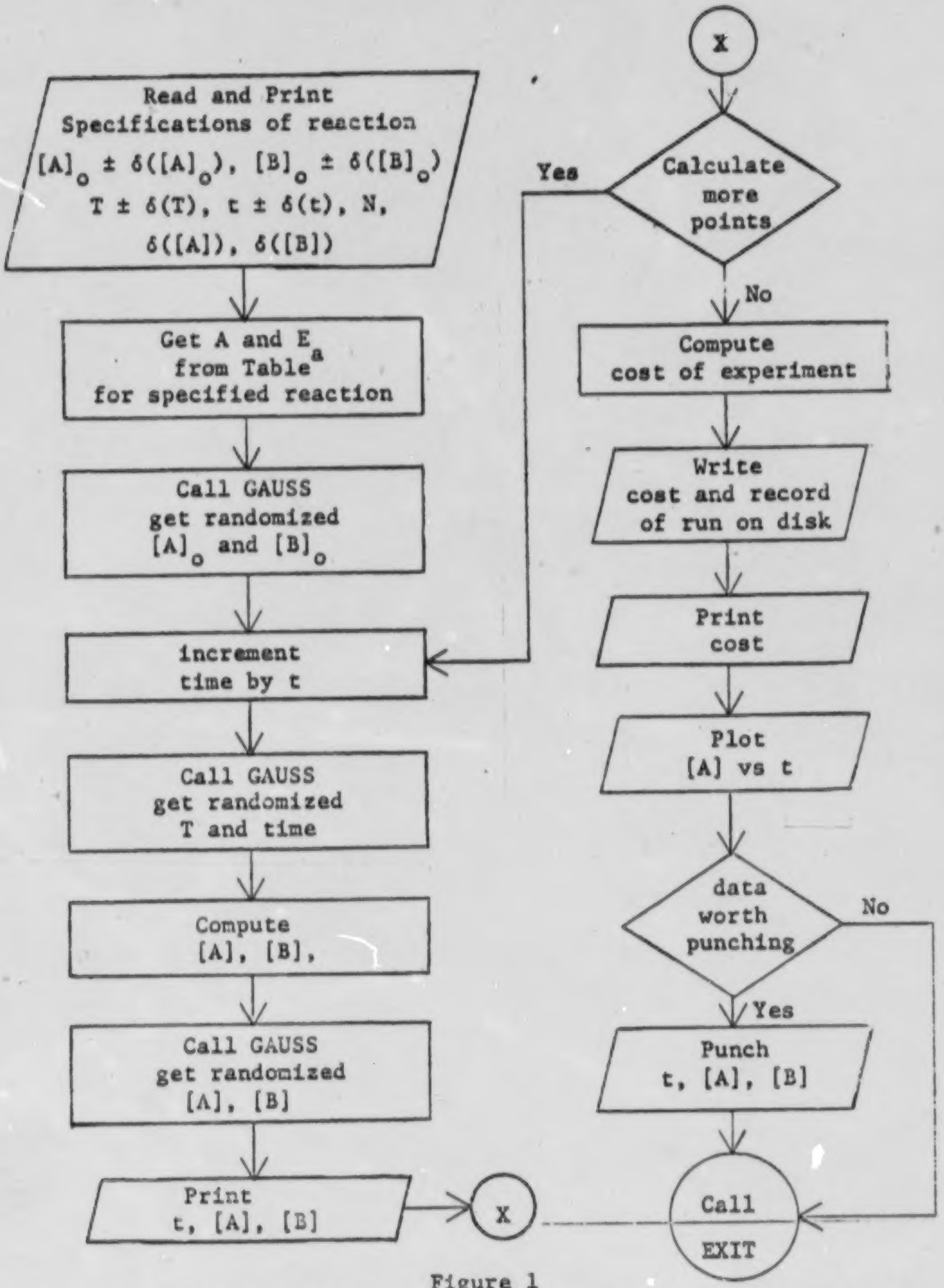

Read and Print
Specifications of reaction
[A]o ± δ([A]o), [B]o ± δ([B]o)
T ± δ(T), t ± δ(t), N,
δ([A]), δ([B])
Get A and Ea from Table for specified reaction
Call GAUSS get randomized [A]o and [B]o
increment time by t
Call GAUSS get randomized T and time
Compute [A], [B],
Call GAUSS get randomized [A], [B]
Print t, [A], [B]
X
X
Calculate more points
Yes
No
Compute cost of experiment
Write cost and record of run on disk
Print cost
Plot [A] vs t
data worth punching
No
Yes
Punch t, [A], [B]
Call
EXIT

Figure 1
Simplified flow diagram of KNEXP: a kinetics experiment gaming program.

6-30

The Vector Space Theory of Matter

by

F. A. Matsen
Department of Chemistry
Department of Physics
University of Texas, Austin

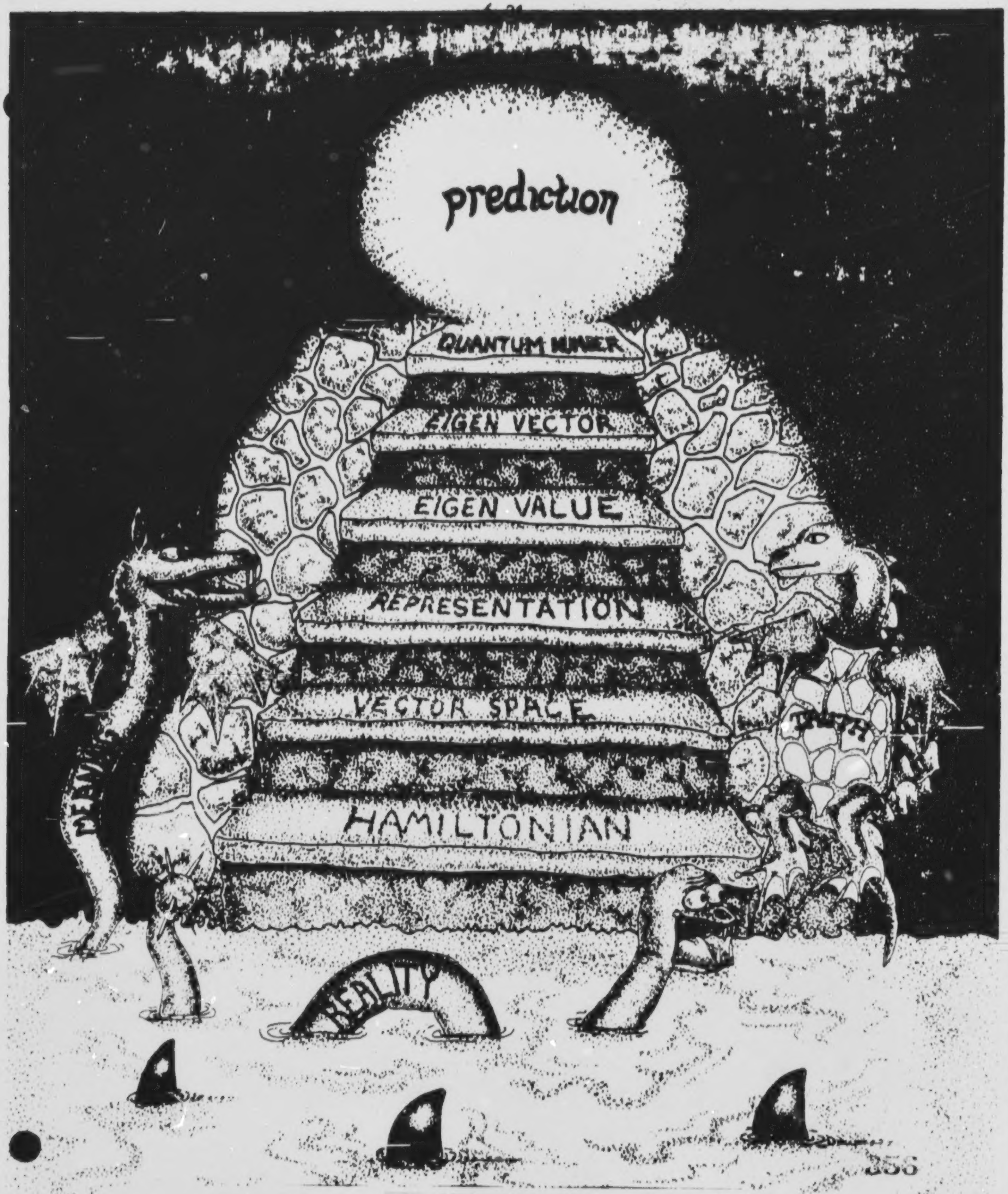

prediction
QUANTUM NUMBER
EIGEN VECTOR
EIGEN VALUE
REPRESENTATION
VECTOR SPACE
HAMILTONIAN
MEANING
TRUTH
REALITY
M. Hansen '72

Figure 1. "Oh, Six-Step Way, lead us from darkness into light, and deliver us from those loathsome beasts who would devour us." (Apologies to Buddha)

I. Introduction

The Vector Space Theory of Matter (VSTM) is the title of a course which has been taught to superior freshman at the University of Texas at Austin for the past five years. It is open to both science and non-science students and satisfies the freshman chemistry requirement. The motivation for the course is as follows:

i. to prevent the acquisition of myths about theories in general and theories of structure of matter in particular,

ii. to shorten the time required for the student to get to the forefront of modern developments,

iii. to teach superior non-science students about scientific theory,

iv. to provide for all students a philosophy of science,

v. to challenge those students who have had a superior high school chemistry course,

vi. to provide a realization of the 'new-math' concepts,

vii. to provide computer orientation in a chemistry environment.

VSTM is concerned with micromatter: atoms, molecules, nuclei, and elementary particles. The science syllabus is given in Table 1. The starred topics constitute a short course of about twenty lectures.

VSTM employs vector space (Dirac-Heisenberg) mechanics which emphasizes vector spaces and operators rather than wave (Schroedinger) mechanics which emphasizes differential equations. The mathematics which is required for VSTM is taught as an integral part of the course and is based on the concept of a set. Three examples of sets are exhibited and exploited. They are:

a. the complex number field,

b. vector spaces over the complex number field,

c. algebras of operators on vector spaces.

VSTM does not require calculus. The syllabus alternates between mathematics and science, a technique which makes for considerable reinforcement of the learning process. The science provides examples for the mathematics and the mathematics provides a realization of the science.

The good student has little difficulty with either the mathematics or its application in VSTM. He does, however, have considerable difficulty with the acceptance of the VSTM because of his association of the objectively real with the classical deterministic world. To reduce the student's philosophical problems, there is also taught as an integral part of the course, the ACP (axiomatic cum pragmatic) epistemology. This philosophical part of VSTM is, for many students, the most exciting part of the course.

According to ACP, the set containing all man's experiences can be subdivided into two subsets: the physical subset and the humane subset. Regarding the physical subset, ACP claims the following:

a. the physical subset of man's experience consists solely of observations and theories about the observations,

b. a theory is a deivce for predicting observations,

c. a theory is judged by the diversity and accuracy of its predictions,

d. concepts like truth, reality and meaning play no useful role in theory evaluation. Historically, the use of idealistic value judgements have been counterproductive for science. Although we are primarily interested in the physical subset, much of the epistemology of the physical subset gets ultimately transferred into the humane subset.

In VSTM all theories of micromatter are formulated in the six-step way (see Figure 1). Questions about terms of reality, truth and meaning are declared off limits. We close this section with a few remarks about step six of the six-step way. Quantum numbers can be inferred from experiment. To predict quantum numbers we must select a Hamiltonian (step 1) which commutes with every element of some associative (Frobenius) algebra or a non-associative (Lie) algebra. On the basis of the Wigner-Eckart theorem, we then predict that the observed quantum numbers are the irreducible representations of the algebra. They are accordingly called *algebraic quantum numbers*. For example, for the hydrogen atom

$\mathcal{L}GR(4) \Rightarrow n$, the principle quantum number

$\mathcal{L}GR(3) \Rightarrow \ell, m$, the azimuthal and magnetic quantum numbers

$\mathcal{L}GSU(2) \Rightarrow s$, the spin quantum number

For elementary particles

$\mathcal{L}GSG(3) \Rightarrow I$ (isospin), Y (hypercharge)

The textbooks for VSTM are:

F. A. Matsen, *Vector Space and Algebra for Chemistry and Physics*, Holt, Rinehart and Winston, 1971.

F. A. Matsen, *Vector Space Theory of Matter*, Vol. I (Chemistry), University of Texas Co-Op (1970); Holt, Rinehart and Winston, (Scheduled 1972).

F. A. Matsen, *Vector Space Theory of Matter*, Vol. II (Physics), Holt, Rinehart and Winston, (Scheduled 1972).

II. Computer Usage in VSTM (with Robert Wesson and Barbara Sapienza)

In VSTM the computer is used in three ways:

a. the laboratory contains an experiment consisting of simple Fortran programming,

b. the eigenvectors and eigenvalues for large systems are computed with canned programs,

c. CAI.

An example of CAI is a tutorial for the computation of eigenvectors and eigenvalues. Given a matrix the computation is carried out as follows:

Step 1. Solve eigenvalue problem,

Step 2. Write eigenvector equation in ratio form,

Step 3. Solve for ratios,

Step 4. Normalize the vector.

The CAI is called a tutorial because it presupposes some previous knowledge on the part of the student which is acquired by the so-called quick and dirty method:

i. a twenty minute lecture followed immediately by a quiz,

ii. a second lecture over the same material followed immediately by a quiz.

Those who scored perfectly on quiz i are excused from quiz ii and whatever follows. Those who scored perfectly on i or ii are excused from what follows. The remaining ~15% are required to take the CAI tutorial.

A flow diagram for the eigenvalue-eigenvector tutorial beginning with step 2 is given in Figure 2, a typical print out in Figure 3 and a glossary in the appendix. The math text to which reference is made in the tutorial is Vector Spaces and Algebras. The tutorial employs as a programming language CLIC (Conversational Language of Instruction and Computing) which is translated and compiled as a Fortran program on the UT CDC6600-6400 system. The program at remote terminal is executed with the time sharing language TAURUS.

Three interesting features have emerged:

a. The quick and dirty method solves many student problems and greatly relieves the load on CAI programming and the computer,

b. The eigenvector-eigenvalue tutorial does more than teach the student eigenvectors and eigenvalues. It also teaches him how to study applied mathematics. Students often become discouraged when they cannot see answers immediately. The CAI tutorial slows them down (it is verbal) and indoctrinates them with the necessity of taking one step at a time and of focussing carefully on each step. Further, it teaches them that questions about truth, reality and meaning are not part of the computation. This lesson carries over and we find CAI is less necessary later in the course.

c. There remains a hard core five percent who do not respond to CAI. Some of these can be handled on a personal tutorial basis but others are psychologically indisposed toward the reduction of scientific concepts to steps in a computation. We try to cycle these people out of VSTM into a conventional course by the end of the first week.

In summary, VSTM makes all students (science and non-science) computer conscious. In fact, it has influenced many students in the direction of computer science.

Table I

Territory Accessible by the Six-Step Way

1. Simple Orbital Theory

   * a. Atomic Orbitals
   * b. Hückel Orbitals
   * c. Ligand Field Orbitals
   d. Bloch (Solid State) Orbitals
   e. Nucleonic Orbtials
   f. Quark Orbitals
   g. Spin Orbitals

2. Coarse Structure Theory

   a. Atoms
   b. Unsaturated Molecules
   c. Complex Ions
   d. Solids
   e. Nuclei (Supermultiplet Theory)
   f. Baryons, Mesons

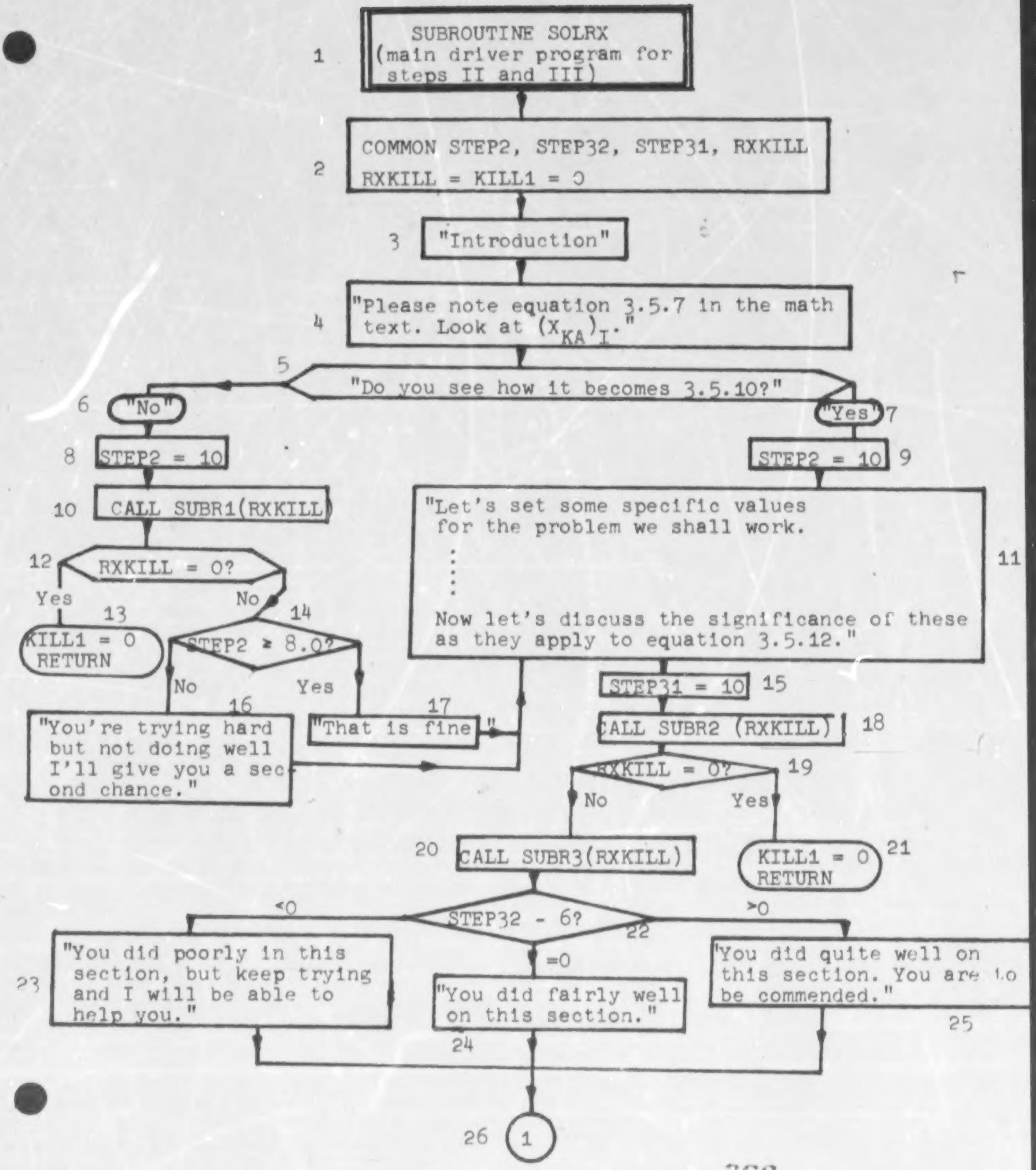
3. Fine Structure Theory

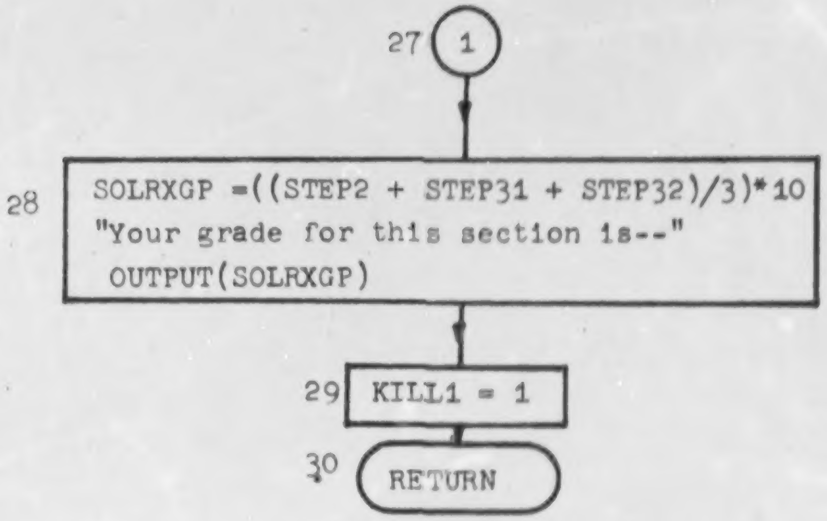
   a. Fine and Hyperfine Structure
   * b. NMR and ESR
   c. Nuclei (Isospin Free Theory)

APPENDIX I

GLOSSARY

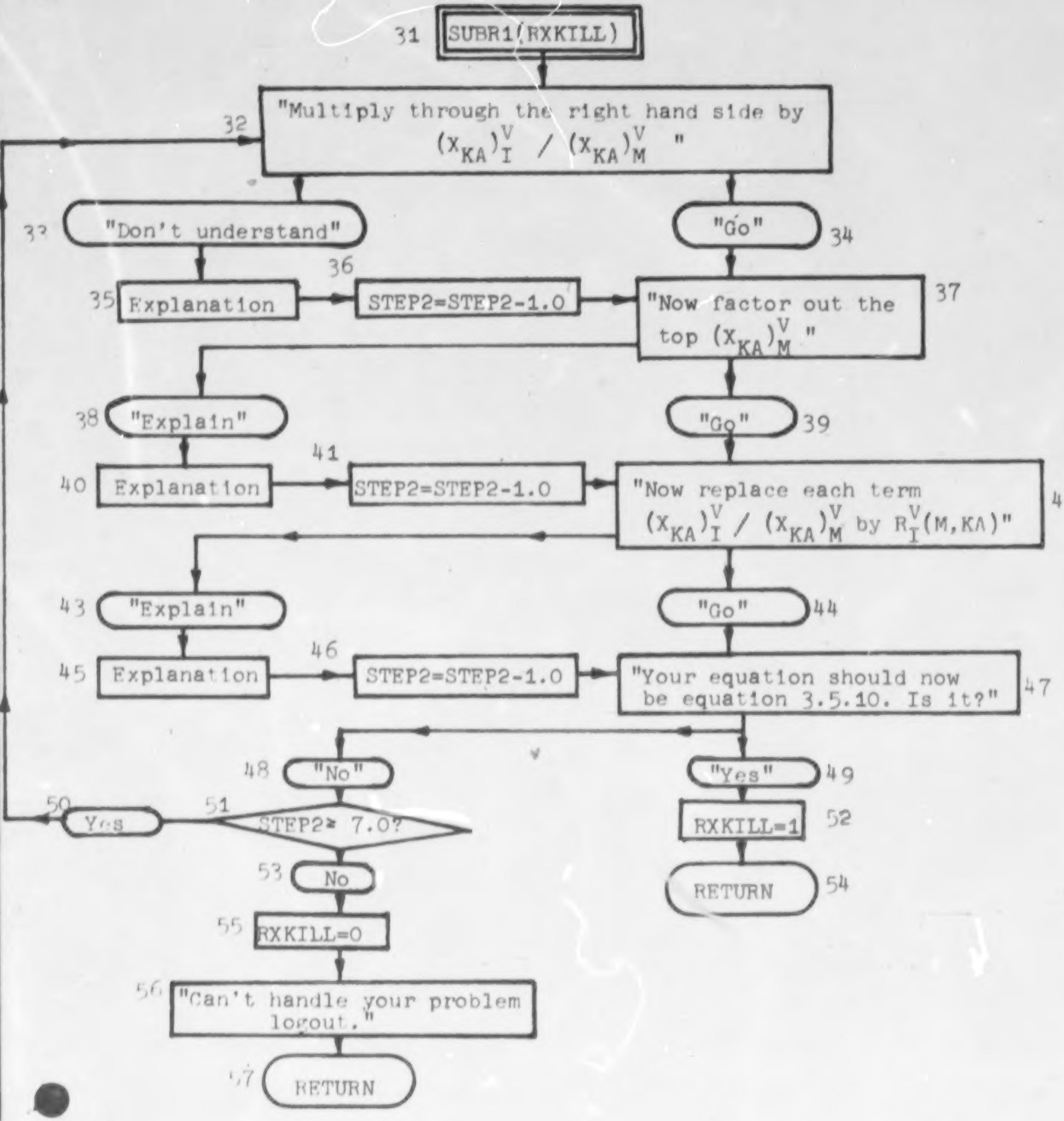
STEP2	initialized in driver program to 10 in SUBR1, decreased 1 for each request for explanation or wrong answer
STEP31	initialized in driver program to 10 in SUBR2, decreased 1 for each answer given wrong twice
STEP32	internal control in SUBR3, works the same way as above
RXKILL	a control to see if the student can perform the work in the subroutines. It is initialized 0 in the driver program. If the student shows that he has been unable to perform the work, RXKILL is set at 0 which signals the driver program to refer him to the proctor and end the tutorial. If the student has done adequately, RXKILL is set to 1 which signals the driver program to continue.
KILL1	used only in the driver program. Calls routine which refers the student to the proctor and ends the tutorial. Signalled by RXKILL
CA	correct answer
CB	alternate correct answer
UN	wrong answer

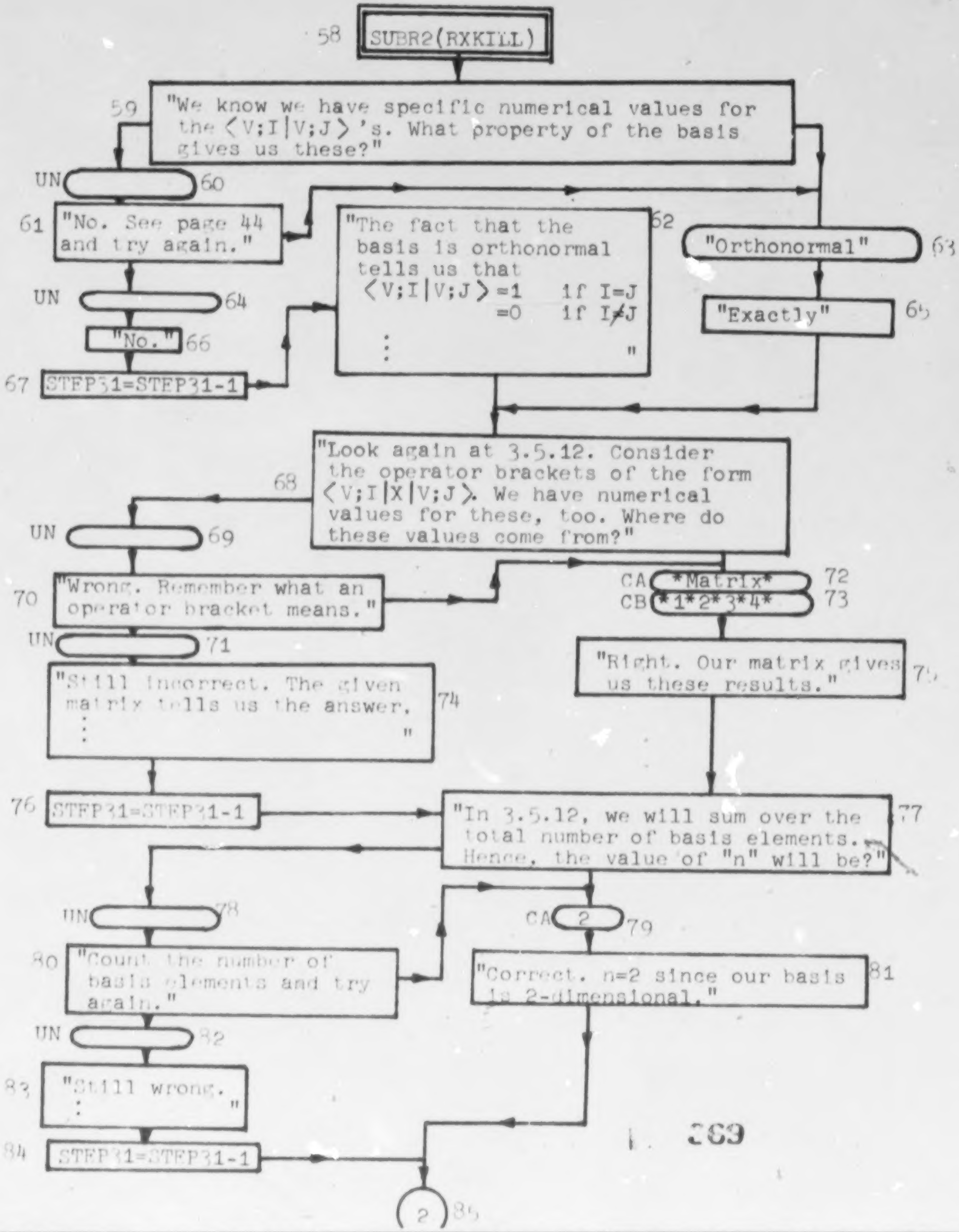

1
SUBROUTINE SOLRX
(main driver program for steps II and III)
2
COMMON STEP2, STEP32, STEP31, RXKILL
RXKILL = KILL1 = 0
3
"Introduction"
4
"Please note equation 3.5.7 in the math text. Look at $(X_{KA})_I$."
5
"Do you see how it becomes 3.5.10?"
6
"No"
7
"Yes"
8
STEP2 = 10
9
STEP2 = 10
10
CALL SUBR1(RXKILL)
11
"Let's set some specific values for the problem we shall work.
⋮
Now let's discuss the significance of these as they apply to equation 3.5.12."
12
RXKILL = 0?
Yes
No
13
KILL1 = 0
RETURN
14
STEP2 ≥ 8.0?
No
Yes
15
STEP31 = 10
16
"You're trying hard but not doing well I'll give you a second chance."
17
"That is fine"
18
CALL SUBR2 (RXKILL)
19
RXKILL = 0?
No
Yes
20
CALL SUBR3(RXKILL)
21
KILL1 = 0
RETURN
22
STEP32 - 6?
<0
=0
>0
23
"You did poorly in this section, but keep trying and I will be able to help you."
24
"You did fairly well on this section."
25
"You did quite well on this section. You are to be commended."
26
1

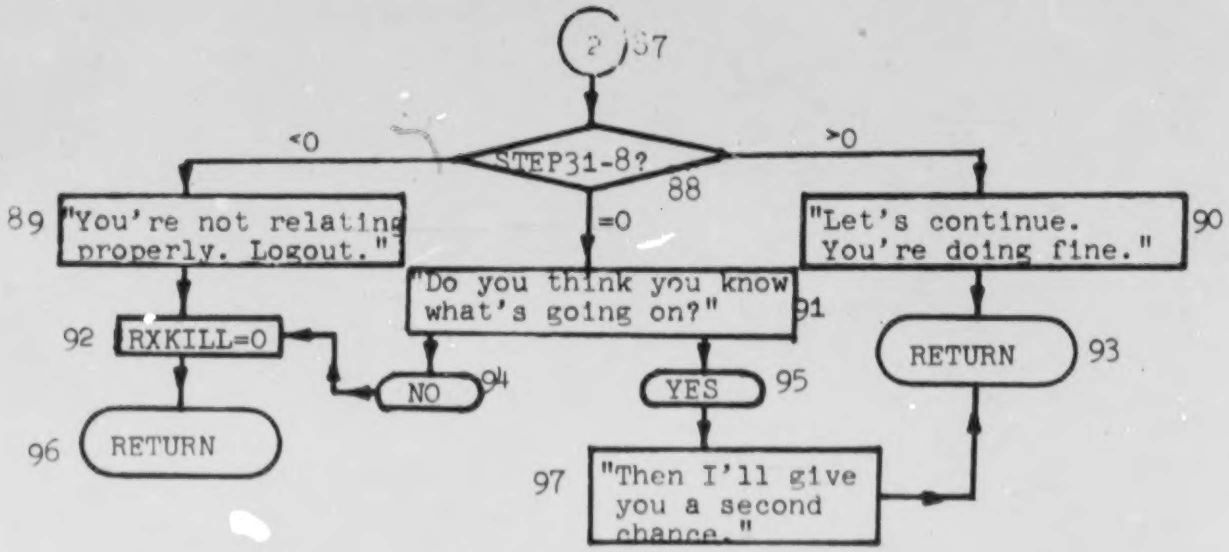

27
1
28
SOLRXGP =((STEP2 + STEP31 + STEP32)/3)*10
"Your grade for this section is--"
OUTPUT(SOLRXGP)
29
KILL1 = 1
30
RETURN

This subroutine is called with two parameters - KILL1 and SOLRXGP. If KILL1 is returned as zero, the student could not adequately perform properly on either it or any of its subordinate subroutines. He should be referred to the proctor. If KILL1 is returned as 1, the student completed the subroutine properly.

The other parameter SOLRXGP is the floating-point grade point average of the student on a 100 point scale. The normal A, B, C, D, and F grades can be assigned according to this scale, which should be a normal distribution for most classes.


31
SUBR1(RXKILL)
32
"Multiply through the right hand side by $(X_{KA})_I^V / (X_{KA})_M^V$ "
33
"Don't understand"
34
"Go"
35
Explanation
36
STEP2=STEP2-1.0
37
"Now factor out the top $(X_{KA})_M^V$ "
38
"Explain"
39
"Go"
40
Explanation
41
STEP2=STEP2-1.0
42
"Now replace each term $(X_{KA})_I^V / (X_{KA})_M^V$ by $R_I^V(M,KA)$"
43
"Explain"
44
"Go"
45
Explanation
46
STEP2=STEP2-1.0
47
"Your equation should now be equation 3.5.10. Is it?"
48
"No"
49
"Yes"
50
Yes
51
STEP2≥ 7.0?
52
RXKILL=1
53
No
54
RETURN
55
RXKILL=0
56
"Can't handle your problem logout."
57
RETURN


58
SUBR2(RXKILL)
59
"We know we have specific numerical values for the ⟨V;I|V;J⟩'s. What property of the basis gives us these?"
UN
60
61
"No. See page 44 and try again."
62
"The fact that the basis is orthonormal tells us that ⟨V;I|V;J⟩=1 if I=J =0 if I≠J
⋮
"
"Orthonormal"
63
UN
64
"Exactly"
65
"No."
66
67
STEP31=STEP31-1
68
"Look again at 3.5.12. Consider the operator brackets of the form ⟨V;I|X|V;J⟩. We have numerical values for these, too. Where do these values come from?"
UN
69
70
"Wrong. Remember what an operator bracket means."
CA
Matrix
72
CB
*1*2*3*4*
73
UN
71
"Right. Our matrix gives us these results."
75
74
"Still incorrect. The given matrix tells us the answer.
⋮
"
76
STEP31=STEP31-1
77
"In 3.5.12, we will sum over the total number of basis elements. Hence, the value of "n" will be?"
UN
78
CA
2
79
80
"Count the number of basis elements and try again."
81
"Correct. n=2 since our basis is 2-dimensional."
UN
82
83
"Still wrong.
⋮
"
84
STEP31=STEP31-1
2
85


2
87
STEP31-8?
88
<0
=0
>0
89
"You're not relating properly. Logout."
"Let's continue. You're doing fine."
90
"Do you think you know what's going on?"
91
92
RXKILL=0
NO
94
YES
95
RETURN
93
96
RETURN
97
"Then I'll give you a second chance."

CAI EVEC OUTPUT

Student responses underlined. Flow diagram box numbers in margin.

```
   YOU ARE NOW GOING TO PERFORM STEPS II AND III OF THE EIGENVECTOR
   PROBLEM AS EXPLAINED IN PAGES 90 - 93 OF YOUR MATH TEXT.
   NOTE EQUATION 3.5.7 AND LOOK SPECIFICALLY AT THE EXPRESSION

        (X   )
4         KA I

   NOTE THAT SINCE I CANNOT PRINT THE LOWER-CASE "K", I AM FORCED
   TO USE THE "A" IN MY OWN VERSION OF THIS EXPRESSION.  PLEASE DO
   NOT LET THAT CONFUSE YOU.
5  DO YOU SEE HOW EQUATION 3.5.7 BECOMES EQUATION 3.5.10 THROUGH
   SIMPLE ALGEBRAIC MANIPULATION?
6  NO      JUMPS TO SUBR1
   THEN A LITTLE EXPLANATION IS IN ORDER.
   ALL RIGHT.  ANYTIME YOU DON'T UNDERSTAND A STEP I ASK YOU
   TO MAKE, TYPE "EXPLAIN".  OTHERWISE, TYPE "GO" WHEN YOU'VE DONE
   THE STEP.  EXPAND EQUATION 3.5.7 (PERFORM THE SUMMATION OPER-
32 ATION AND WRITE DOWN THE TERMS ON YOUR PAPER).  NOW MULTIPLY
   THROUGH ON THE RIGHT-HAND SIDE BY THIS EXPRESSION (WHICH
   EQUALS ONE)--
             V        V
        (X )   / (X )
          K M      K M
   I'LL WAIT FOR YOU TO DO THIS....
34 GO
37 NOW FACTOR OUT THE (X )    IN THE NUMERATOR....
                          K M
39 GO
   NOW REPLACE EACH TERM OF THE FORM
               (X ) /(X )
42               K I   K M
   WITH THE EXPRESSION     K (M;K)
                             I
   THIS IS JUST A CHANGE OF NOTATION....
43 EXPLAIN
   EQUATIONS 3.5.10 AND 3.5.11 SHOULD GIVE YOU ALL THE INFOR-
45 MATION YOU NEED.  EQUATION 3.5.10 IS PRECISELY WHAT YOUR EQUATION
   SHOULD NOW LOOK LIKE.  DON'T WORRY ABOUT THE VALUE OF M YET.
   PLEASE MAKE THE NECESSARY CHANGES NOW AND LET'S CONTINUE....
47 YOUR EQUATION SHOULD NOW BE EQUATION 3.5.10
   WITH THE LOWER-CASE "K'S" SUPPRESSED.  IS IT?....
49 YES       RETURNS TO SOLRX WHICH CONTINUES TO SUBR2
```

```
SUBC2 AS WE MOVE INTO STEP III, EQUATION 3.5.12 IS THE IMPORTANT ONE.
      DON'T LET IT SCARE YOU.
      IN ORDER TO WORK THE PROBLEM IN AS CLEAR A WAY AS POSSIBLE,
      LET'S SET SOME SPECIFIC VALUES WE WANT TO USE THROUGHOUT THIS
      PROBLEM---


        1)  OUR BASIS WILL BE--      IV;1>,  IV 2>
        2)  WE WILL HAVE NO NEED FOR THE "A" INDEX (LOWER-CASE "K" IN YOUR
            BOOK), SO I WILL SUPPRESS IT.
        3)  M = 1  (YOU'LL LEARN LATER WHY)
        4)  THE BASIS WILL BE ORTHONORMAL.
   59   5)  THE SPECIFIC EXAMPLE MATRIX WE USE WILL BE--

                 3   2

                 2   3


      LET'S DISCUSS THE SIGNIFICANCE OF THESE AS THEY APPLY TO EQUATION
      3.5.12.
      WE KNOW WE HAVE SPECIFIC NUMERICAL VALUES FOR THE <V;IIV;J>
      SYMBOLS. WHAT INITIALIZATION CONDITION GIVES US THESE?
   63 ORTHONORMAL BASIS
   65 EXACTLY
      LOOK AGAIN AT EQUATION 3.5.12. CONSIDER THE OPERATOR BRACKETS OF
   68 THE FORM <V;IIXIV;J>. WE HAVE NUMERICAL VALUES FOR THEM TOO.
      WHERE DO THESE COME FROM?....
   72 GIVEN MATRIX
   75 RIGHT. OUR MATRIX GIVES US THESE RESULTS.
   77  IN 3.5.12 , WE WILL SUM OVER THE TOTAL NUMBER OF BASIS ELEMENTS.
      HENCE THE VALUE OF "N" WILL BE WHAT?....
   78 THREE
   80 NO. COUNT THE NUMBER OF BASIS ELEMENTS AND TRY AGAIN.
   79 TWO
   81 CORRECT. N=2 SINCE OUR BASIS IS 2 DIMENSIONAL.
      LET'S CONTINUE. YOU'RE DOING FINE.    RETURN TO SOLRX WHICH CONTINUES TO SUBC3
```

Introduction to Computers and Computing in the Physical Chemistry Laboratory

M. R. James and W. G. Koch, Department of Chemistry
University of Northern Colorado
Greeley, Colorado 80631

Abstract: A laboratory course has been developed which introduces the beginning physical chemistry student to computers and computing by considering (1) how computers handle data, (2) basic information on computer hardware and software, (3) levels of languages by which programmers can communicate with the computer, and (4) programming and treating data with both programmable calculators and computer systems. The course begins with an introduction to binary numbers and how digital computer operations are based on two-state devices. A brief introduction to computer hardware and software is then presented followed by a consideration of the application of a binary number system to communication with the computer in machine language. This is then followed by an introduction to assembly language with emphasis on how these "low level" languages allow communication in a step-by-step process on or near the "computer level." Emphasis is also placed on how much more difficult these "low level" languages are to learn and use than the "high level" languages. Concurrent with the above, instruction and experience in using programmable calculators is given. Following the above material, instruction in programming in a "high level" language may be given depending on the background of the students and computer facilities available. The latter skills are then applied during the remainder of the year to treating experimental data acquired in the physical chemistry laboratory.

Introduction to Computers and Computing in the Physical Chemistry Laboratory

M. R. James and W. G. Koch, Department of Chemistry
University of Northern Colorado
Greeley, Colorado 80631

In organizing a laboratory course in physical chemistry for use during the first few weeks of the year, one is faced with designing lecture materials and/or experiments which are meaningful and yet do not require a significant background of knowledge of physical chemical principles. One way this may be achieved is by initially lecturing on such subjects as treatment of experimental data, safety in the laboratory, etc., followed by performing fundamental experiments such as Calibration of a Volumetric Flask, Gas Thermometry, Determination of the Molecular Weight of a Volatile Substance by the Dumas Method, etc. An alternate approach which has been tried and found successful is to utilize the laboratory during this initial period to introduce the students to how computers operate and how they may be used in treating experimental data obtained in the laboratory.

With the increase in the application of computer technology in industry and with the increased use of computers in secondary school curriculum, chemistry majors graduating with a bachelor's degree under a conventional college chemistry program often find that they are poorly prepared to fit into existing programs which utilize computers or lack the ability to initiate the use of computers in their respective occupations. In recognition of this problem a program has been developed for use in the physical chemistry laboratory which introduces the students to the basic concepts of how computers operate and how they and programmable calculators may be used to treat experimental data. This program provides a meaningful experience during the first part of the year while the students are acquiring a background in physical chemistry in the lecture portion of the course. Applications to various experiments performed in the laboratory and problems assigned in the lecture may then be made throughout the rest of the year.

The laboratory course is outlined by the following schedule with appropriate variations being used depending on the student's background and interests.

Laboratory 1 - Binary Number Systems and Keyboard Programming of Programmable Calculators

I. Number Systems: 1 to 1½ hours
- A. Digital computers as two-state electronic devices
- B. Generalized decimal system
- C. Binary number system
- D. Conversion of binary to decimal numbers and vice versa
- E. Octal notation
- F. Computer accuracy and word size
- G. Binary arithmetic
  - 1. Addition
  - 2. Two's complement and subtraction
  - 3. Multiplication and division through addition and subtraction

II. Keyboard Programming of Programmable Calculators: 1½ to 2 hours
- A. Instruction and demonstration of basic concepts of programming available calculator(s)
- B. Basic mathematical operations in keyboard programming
  - 1. Data entry group
    - a. Decimal numbers
    - b. Special numbers, eg. e, π, etc.
  - 2. Algebraic group
    - a. Addition, subtraction, multiplication, division
    - b. Exponentiation
  - 3. Special function group
    - a. Reciprocals, factorials, etc.
    - b. Square roots, squares, etc.
    - c. Trigonometric functions
    - d. Logarithmic functions, etc.
- C. Demonstration of simple programming
  - 1. Obtaining averages
  - 2. Calculation of course percentages
- D. Student programming of simple problems
  - 1. Summing a series of numbers
  - 2. Conversions between temperature scales

Laboratory 2 - Computer Hardware and Software and Card or Tape Programming of Programmable Calculators

I. Computer Hardware: 1½ to 2 hours
- A. Characteristics of general-purpose digital computers
- B. Basic elements of a computer
  - 1. Input and output sections
  - 2. Arithmetic unit
  - 3. Memory
  - 4. Control unit
  - 5. Types of words

- C. Basic machine operation
    - 1. Fetch phase
    - 2. Execute phase
- D. Core memory
- E. Computer instruction set
    - 1. Memory reference instruction
    - 2. Register reference instruction
    - 3. Input-Output instructions
- F. Input-Output Systems
    - 1. Teleprinter input/output
    - 2. High-speed punched tape input
    - 3. Card readers
    - 4. High-speed punched tape output
    - 5. Magnetic tape input/output and other peripheral devices where appropriate

II. Computer Software

- A. Classes of software
    - 1. Translators
        - a. Assemblers
        - b. Compliers
- B. Control systems
- C. Utility routines
- D. Applications programs

III. Card or Tape Programming of Programmable Calculators: 1 to 1½ hours

- A. Instruction and demonstration of programming the calculator using cards or tapes
    - 1. Preparing the card or tape
    - 2. Entering the program into the calculator
- B. Storage control unit
    - 1. Storing and recalling data
    - 2. Special storage registers
        - a. Simple accumulating registers
        - b. Statistical summations, etc.
- C. Student programming using cards or tapes
    - 1. Preparation of cards or tapes for programs assigned in laboratory I
    - 2. Application of techniques to original problems of the student's choosing

Laboratory 3 - Machine and Assembly Languages as a Means of "Communicating" with the Computer and Advanced Programming of Programmable Calculators

- I. Machine and Assembly Languages: 1½ to 2 hours
  - A. Review of binary number system and the means by which computers utilize this system
    - 1. Demonstration of data numbers in machine language
    - 2. Memory locations represented in machine language
    - 3. Examples of types of instructions represented in machine language
  - B. Sample program in machine language
    - 1. Development of program to add two numbers and subtract a third
    - 2. Storing the data and program in memory
    - 3. Executing the machine language program
  - C. Sample program in assembly language
    - 1. Translation of the above program into assembly language using appropriate computer mnemonics
    - 2. Storing the data and program in memory
    - 3. Executing the assembly language program
  - D. Review of translators and the use of "high level" language in communicating with the computer
    - 1. Difficulty of programming in machine and assembly languages
    - 2. Concept of "high level" languages
    - 3. Illustration of how the above program would be written in an appropriate "high level" language
- II. Advanced Programming of Programmable Calculators: 1 to 1½ hours
  - A. Practice in programming mathematical problems
    - 1. Sine of an angle given in radians
    - 2. Real roots of a quadratic equation, etc.
  - B. Application of programming to calculations in physical chemistry
    - 1. Boltzmann distribution of particles in a given set of energy levels
    - 2. Heat of combustion for bomb calorimeters, etc.

Laboratory 4, etc. - Appropriate Instruction and Application in Programming a Computer in a "High Level" Language: Number of lab periods depends on the background of the students

The language chosen depends on the background of the students and on the computer facilities available. If a simple language such as BASIC is chosen then a fairly sophisticated degree of programming skills can be acquired by the better students in a relatively short period of time. For the case where a relatively difficult language such as FORTRAN IV must be used, no attempt is made in the laboratory to give detailed instruction in this language. Rather, a brief exposure (no more than 3 laboratory periods) is given and the students are encouraged to enroll in an existing course in FORTRAN IV sometime during the year and then to apply this knowledge in their laboratory and course work.

In addition to the rationale given earlier it is felt the above approach is beneficial to the students from several other points of view. One goal of the course is to prepare the student to treat data in a programmed fashion in a bi-level approach: (1) programmable calculators and (2) full-scale computers. The material on calculators was included because of the immediate applications in physical chemistry and also they may be the only programming tool financially within reach of some of the students after they leave school. This is particularly true for those employed by small schools or businesses. The material on computers was prepared with the goal in mind of giving the students a broader exposure than is typically available in the standard course in programming.

For the student who is a teaching major the course is designed so as to enable him to work with students who are exposed to computer programming in their high school math courses and also to give him a more fundamental knowledge of computer technology which will assist him in giving the students a greater breadth of knowledge about computers in general.

For the industrial or research chemist the course not only provides an introduction to the treatment of data by computer techniques but also gives them an introduction to some of the basic knowledge necessary for programming computers for on-line data acquisition from chemical instrumentation.

In operating such a course a great deal of printed matter is often available from computer firms, local telephone companies, etc. These can usually be obtained either free of charge or at a minimal cost.

The Use of a Teaching Computer in the Undergraduate Laboratory

C. E. Klopfenstein
Department of Chemistry
University of Oregon
Eugene, Oregon 97403

C. Wilkins
Department of Chemistry
University of Nebraska
Lincoln, Nebraska 68508

We began implementation of our "Distributed Computer Network" in early 1969 through generous NIH funding. Phase I of this program involved acquisition of three laboratory computers with characteristics similar to those presented in the paper read by Dr. Wilkins. After acquisition of the system, our first task was to begin an educational program at all levels to familiarize our undergraduate, graduate students and faculty with the mechanisms by which the computers could be used to augment on-going research projects. An adequate test-laboratory manual to support such a program could not be found so a collaborative effort to develop the necessary teaching materials was initiated between the authors of this paper. A serious effort in this direction has been made possible through the generous financial support of the National Science Foundation.

In the initial stages of the program, we discovered that over 95% of the small computer machine time was being used for the editing of paper tapes and assembly of programs from paper tape. This left little time for program execution, and greatly reduced mean time between failures for our teletypes since activities of the above type require extensive hard copy. Part of the teletype problem was solved by writing an oscilloscope based interactive editor program, but the time element still was unfavorably weighted towards program development and away from experiment development.

We then developed an assembler program that would function within the University supported computer system, but would produce code that could be executed using the laboratory computers. The disadvantage in that some of the small computer "hands on" experience has been transferred from editing assembly to only actual execution is far outweighed by the facts that programs are punched on easy to edit card decks and that all students have access to the assembler at all times. To augment the power of our "off line" assembler, we have developed a simulator (written in FORTRAN) that can utilize a large computer to debug certain kinds of errors in the small machines programs. At least one commercial timesharing house offers a similar assembler-simulator package that operates to make any terminal attached to the system appear as though it is attached to a small computer. It is, of course, difficult to simulate the random actions of a real time environment. However, now that the lengthy process of assembly is removed from the smaller system, much more time is available for program and hardware testing. To facilitate the latter, we have written an interpretive execution package that allows for complete tracing of program execution, and provides the students with debuging information that would be difficult to obtain in any other way. A program is executed while the step-by-step status of the small machine is provided to the student in hard copy. Operation of programs using the interpreter has greatly enhanced the student's understanding of the small machines' instruction set.

The type of hardware described in the previous paper has proven to be ideally suited for interfacing experimental apparatus to the computer in both the research and educational environment.

The isolation provided by our optically coupled analog-to digital converters and the programmed gain features of the front end have proven invaluable to the interfacing of our equipment, since typically no special grounding considerations were incorporated into the design of the instruments the students have available for interfacing.

Some of the experiments are performed using ready made interfaces and already written programs. These generally involve some particularly complex problem such as simulation of Nuclear Magnetic Resonance or Electron Spin Resonance Spectra. The educational value of such simulations has long been recognized, but in most instances only large batch type computers are used where the turn around time may be hours. Using a 4K-16 bit computer equipped with only a teletype, and DAC, it is possible to simulate 5-spin Nuclear magnetic Resonance spectra and have the results plotted instantly on any standard voltage-vs-time recorder. Programs of this type have been utilized in both our qualitative organic analysis and physical chemistry laboratory courses. In the former, students are given unknown compounds with complex but interpretable NMR spectra which when properly analyzed, could give the student helpful structural information. The student determines the NMR spectrum of his unknown and then using the simulator, interprets the spectrum in terms of coupling constants and chemical shifts, which in turn he interprets in terms of his proposed structural assignment. Students in Physical Chemistry Laboratory perform similar experiments, except here the design is to teach something about the quantum chemistry of the technique. One experiment involves observing changes in NMR line positions and intensities due to changes in solvent polarity. The changes arise due to induced degeneracy of chemical shifts causing normally forbidden lines to become allowed and vice versa. In fact, it is possible to determine the relative signs of coupling constants between some of the protons in the compounds used. Of course, interpretation of the data requires full calculation of several spectra, and when the student is able to immediately observe the effects of modifying one parameter of another, his understanding of the physical process involved is greatly enhanced, and the time required for the experiment is reduced over that if the simulations were performed using a batch computer. Experiments have also been performed utilizing electron paramagnetic resonance spectroscopy simulation with similarly encouraging results. Besides the chemistry taught by these experiments, the "hands-on" aspects add to the students understanding of computer technology.

Another category of experiments that helps to introduce students to digital data acquisition makes use of the general interface and a high level language for data acquisition and data reduction. The languages used in our case are FORTRAN, BASIC, and CLASS, but similar languages are available for us with most small computers. By nature, these systems are limited in acquisition rates and real time computational ability, but a great bulk of the experiments performed in undergraduate laboratories are slow and relatively undemanding in control requirements. Examples of such experiments are spectrophotometrically monitored kinetic experiments, automatic spectrophotometer mixture analvsis, microdensitometer control, vapor phase chromatography experiments, and several other experiments that produce data as some regular function of time or voltage. Examples of such programs are available from the author on request. The main advantage of this or any other high level language is that students can perform useful experiments without having to learn much about either the assembler language of the computer or the electronic characteristics of the interface. This means that one or two experiments that utilize the computer can be made to be part of an existing course without completely revamping the emphasis and basic course context. Another important advantage of starting with a high level language is that the "activation energy" of actually learning how to use a computer is greatly reduced over that imposed if the student must first learn an assembly language and interface electronics. It has been our experience that exposure of the above type actually serves to stimulate interest in the more basic details of program generation and instrument interfacing.

A variety of other approaches have been explored in our laboratories, and perhaps the above will be replaced in the near future with some other teaching technique. This term approximately 40 students in a variety of disciplines are participating in one way or another in our computer educational program.

In conclusion, we wish to note that even though this program is still in its developmental stages, nearly all of the plan outlines in paper I have proven to be workable, and after another year we hope to be able to report the implementation of a fully debugged undergraduate teaching curriculum.

Acknowledgements:

The authors wish to acknowledge the generous support of the National Science Foundation through grants GJ 393 and GJ 441.

CHEMISTRY, COMPUTERS & GRAPHIC INTERACTION

Harvey Z. Kriloff
Computer Center and Department of Chemistry
University of Illinois at Chicago Circle
Chicago, Illinois 60680

It is not very suprising that Chemistry, whose roots are located deep in Alchemy, should retain a high degree of graphic partiality. Words to the alchemist served more to confuse than to clarify, while many of the diagrams of chemical apparatus which appear in these old manuscripts (20) would be acceptable in a modern text on Chemistry. Throughout its development Chemistry has depended on the graphic image to clarify concepts (16), and to create theoretical models (9). In some cases graphic devices were necessary before the results of experiments could be clearly understood (12,42). Thus, the modern chemist, in using the computer, has been handicapped because he tries to communicate in an exclusively textual mode. Ideas formulated pictorially, whether in the Chemist's mind or on paper, are most easily communicated graphically (17).

The advantage of clear, visible displays has long been recognized by those who were responsible for demonstrating experiments that were not visible to more than a few people at a time or effectively communicating with large audiences. Very ingenious methods have been developed for increasing the graphical visibility of such experiments (19), as well as effectively exploiting the advantages of modern audio-visual equipment (2). There is no reason why these techniques should not be exploited, rather than ignored, when the chemist uses the computer in either education or research.

It would be helpful at this point to define the term "Computer Graphics". We refer to the ability of the computer to display line drawings, shaded pictures and graphs, under computer program control. These images may appear on the face of an electron tube, on film, on paper or any other type of material which is user readable. We also classify as "Computer Graphics" the study and use of devices for the input of graphical data to the computer, such as digital scanners, light pens, Rand tablets, joysticks, etc.

There is no good reason why the chemist has not made better use of this method of communication. In 1963, Ivan Sutherland (48) and his colleagues, designed and implemented a system that was the first significant demonstration of the use of a graphic display console as an interactive input/output device. This system, called SKETCHPAD, was designed for applications related to computer-aided design, however it effectively showed the utility of such a device for a wide range of computer oriented problems. The graphics console had the dual function of serving as an input/output device for pictorial and aphanumeric data, as well as controling the sequence operation of the computer program. The advantages and past accomplishments of such devices are what we will explore in this paper. However, before we examine the advantages of "Interactive Graphics" we should explore the field of "Passive Computer Graphics".

Those input devices which operate on static types of data, and those display devices which generate irreversible images (once a line has been drawn it can not be erased) are classified as "Passive Graphic Devices". These devices, of which line printers, plotters and photographic devices are representative, are also refered to as "hard-copy" output devices.

Many computer users have discovered that the line printer can be very effectively used as a plotter (10,39,44). While the precision is limited by the size of a print character, this device produces plots that are crude, but adaquate for such things as histograms, line spectra, scatter diagrams, and as masks for hand drawn contour diagrams.

Digital incremental plotters allow the user to employ commands that control the movement of a pen over a sheet of paper in any one of eight different directions while the pen is either raised above or touching the paper surface. Software that uses these "incremental" commands to generate approximations to straight lines (vectors), is all that is really needed to use such a system (5,35). However, in practice, more sophisticated software is usually required (23,52). The size of the smallest possible incremental movement, an item that determines the precision of plotted output, is a major factor in the cost of the plotter. Flat-bed plotters are usually more expensive than the cylindrical type and can often handle larger plotting surfaces. Remote plotting terminals are now available which can be placed at the same location as an interactive alphanumeric terminal to provide a highly satisfactory, more interactive type of graphic display.

The final type of passive graphic display is one that uses a camera and film assembly to record the display. The major devices in this class are units designed for microfilm output, that will also accept 16mm, 35mm, 70mm or 105mm film. These units, called Computer Output Microfilm (COM) devices, are new, quite expensive and only available at a few installations (11,26,34,37,45). However, many of these units operate off-line from a computer, accepting their commands from a computer generated magnetic tape. Thus, at installations where such devices are not available, tapes could be generated using standardized software and then printed at a service bureau type organization. Such a service is now being created at Brooklyn Polytechnic Institute (6).

Among new hard-copy devices that will probably of their ability to provide a more effective man-machine interface. They communicate in graphic mode, a universal shorthand language that can be used to compactly represent the very large volumes of information that the computer is capable of generating. However, they operate in what is essentially a batch environment, with the usual long delay between stimulus and response. Interactive graphic devices allow for more natural communication with the computer, where the user can handle those aspects of the problem where good computer algorithms do not exist. These interactive devices exist in three different configurations, with a wide range of system response characteristics and prices. The configurations start with the least expensive, consisting of a stand-alone dedicated computer that shares its memory with the graphic display unit. The intermediate configuration is one in which a seperate buffer is available to handle the display information. The most expensive, highest performance system consists of a large computer in a network with several smaller computers, where the satellite computers provide the display formatting function, and the large computer provides the large-scale, sophisticated numerical calculation and bulk secondary storage capability.. This is the best perfomance system, with quick response to highly complicated requests. All of these systems are presently quite expensive, but with some modification they can provide a high degree of graphic interaction at minimal cost. The statement, often made in the past, that most users could not afford computer graphics because the equipment was too expensive and the software too complicated (54) is really no longer true, and the converse is probably more correct, that one can't afford not to use computer graphics, just as one can't afford not to use the computer.

The most sophisticated of the three types of interactive display devices is the Cathode Ray Tube (CRT), where under computer control an electron beam traces the displayed image on a light emitting phospher, rewriting the image at a rate that will maintain the light level. This operation known as "refreshing" must be performed from 15 to 60 times per second in order to maintain a "flicker-free" display. This constant regeneration of the image on the tube face is the reason for the high demand of the display on computer resources, as well as the reason why one can create dynamic images on the tube face. Units based on this type of system are still quite expensive, but where available they have proven quite productive

(1,7,24,28,29,40,47,56)

A second type of display unit is the Direct View Storage Tube (DVST), where the unit internally mantains the image placed on the face of the tube for many hours. This type of display therefore requires much less of the computer's resources, and can therefore share a computer among many such graphic displays (14,15). The graphic image is not as dynamic as with the CRT, since with most tubes the complete image must be erased before a line may be moved. This is gradually changing as new storage devices are developed, among which is the "Plasma Panel" created as part of the University of Illinois's Computer Assisted Instruction Project, Project PLATO (4). Both the CRT and the DVST are called "Calligraphic" or "Random Scan" display devices, because the writing element moves across the face of the screen in a random fashion under program control. The third type of display device operates in a different manner.

This device is called a "Raster Scan" display and operates on the same principles as the common television console. This unit must also be refreshed, but it will cover every spot on the tube face with each sweep. The information which must be supplied to this type of display device is whether the electron beam should be on or off at any point in the scan. This device, while requiring a wider bandwidth of information to support the refresh rate, takes advantage of a mass produced technology that provides a display device (a television set) at a very low price (32). These devices can also take advantage of the closed circuit television systems that are being developed and used on many college and university campuses (22,36).

The last two classes of devices provide computer graphics at a very reasonable price. Both types of units can provide inexpensive storage of the computer generated image within the display device, permitting a single computer to drive many graphic consoles. What then are some of the possible applications of these graphic display devices to chemical problems? The easiest answer is to observe some of what has already been done.

In the area of Molecular Model Building and Manipulation some of the most general graphic handling systems have been devised. This is an area in which all branches of Chemistry have a common interest. Much of our understanding of chemical principles has been obtained through the creation of intricate models of chemical systems and we have often found it necessary to build elaborate devices in order to obtain the information we desired from the available models (31). Still many chemists find fault with the available models (43) and create new more intricate models that will display the information they desire. The computer provides a useful facility for building models because most of the available computer graphics systems contain commands which are very similar to the type of operations which one would like to perform on molecules (rotate, translate, move point, etc.). Systems have been designed to assist in the synthesis of complex organic molecules, through their computer graphic manipulation (7,56), and others to help in the understanding and construction of large biochemical models (28,29,42). Such systems allow the user to control the complexity of the system while retaining the full mathematical model internally in the computer.

In a similar manner, computer graphics has been used to aid in the identification of molecules for computerized *Information Retrieval* systems. Graphic units have been used both to aid in the input (24) and display (54) of chemical data. In terms of the elimination of user errors, such interfaces are unmatched.

The most interesting of the various graphic systems are those involved in the *Simulation* of the real world. Such a system can be used to either teach or validate a chemical theory under ideal conditions of control (47). Simulations can be conducted with or without simplifying approximations, as a method of validating the accuracy of such approximations (13). Experiments can also be conducted, using the computer to generate the data, where it is not possible to actually perform an experiment due to the high cost, lack of equipment, or shortage of time (10,14,46). Data can also be obtained from "theoretical" experiments, such as obtaining pictures of molecular orbitals (41), as well as experiments performed under "impossible" experimental conditions (44). The application of graphics provides data in a form similar to that obtained from actual experiments, as well as in a compact and readable format.

A common problem in the generation of such simulations is that the calculations necessary for the simulation can not be performed in "real-time". One solution to this is to use an *Analog Computer* to generate the data (33). In cases where the problem can be expressed as differential equations, such as in kinetics (18) or quantum theory, solutions are easily programmed and demonstrated. Due to the parallel and repetitive architecture, a normal laboratory oscilloscope can function as the display device. Rapid experiments performed in this fashion are very useful as teaching tools (8,49), where many parameters may be adjusted and the system observed under a variety of conditions.

Another solution to the time frame problem is to produce *Computer Generated Films*. The advantage of such films in terms of exactness, clarity and educator control has been expressed elsewhere (21,26,45,57).

These films do not have to be of professional quality (although many are as good or better), they may even be made by students in a laboratory (30), they must simply make the desired concepts clear and interesting. Examples of films of a chemical nature that have been made by such a process are ones showing the vibrational states of a molecule (34), the behavior of molecular orbitals during chemical reactions (50), and the dynamics of molecular collisions (55). A number of Computer Movie Languages have been developed (51,53) to assist in the production of similar movies. Films of this nature can also help us understand the results of phenomena that occur at very high or very low speed, or that occur in regions where we can not observe them (37,38).

A new era in education can be opened for both the student and the teacher when the computer is brought into the Classroom as an Interactive Display device. Accurate 3-dimensional representations, including stereographic and perspective images, are possible through the mathematics of geometry. The teacher is no longer restrained by the limits of his artistic talent, but by his imagination in the creation of effective displays. Whether the system is used for classwide display, or as part of a Computer Assisted Instruction system, the instructor is provided with a new dimension in educational technology. He can provide more accurate direction, even with large classes, in the areas of student interaction (3), manipulative experiments (22), and the class educational pace (36). As yet only the surface has been scratched in this important application of computer and graphics participation, but the future is very encouraging.

A final application of graphics is in the area of on-line Acquisition, Control and Display of Data obtained from chemical experiments (15). As examples, such techniques have or will be applied to x-ray diffraction (40), gas chromatography, mass spectroscopy, NMR, ESR and a variety of other instruments. The graphic form of the display allows the experimenter to accurately monitor and make intelligent decisions about the experimental system. Keyboards, switches, lights, and other one-dimentional indicators do not provide sufficient data for complicated experimental systems.

The above has been only a partial look at what has and will be done with Computer Graphics. This will be an area of intense development over the next few years, and no chemist can afford not to be aware of and to participate in this exciting field of experimentation.

REFERENCES AND SOURCES

1. R.M. Baecker, "Picture Driven Animation", AFIPS Spring Joint Computer Conference, 273-296 (1969)

2. W.R. Barnard, et. al., J. Chem. Ed. 45, 63-70, 136-140, 341-346, 543-546, 617-620, 681-684,745-749, 46, 325-327, 461-464, 47, 318-321 (1968-1970)

3. J.W. Barnes, J.H. Bedenbaugh and F.W. Brown , "The use of Programmed Lectures in Freshman Chemistry", J. Chem. Ed. 46, 622 (1969)

4. D.L. Bitzer and H.G. Slottow, "The Plasma Display Panel", AFIPS Fall Joint Computer Conference, 541-548 (1966)

5. W.T. Bordass and J.T. Linnett, "A New Way of Presenting Atomic Orbitals", J. Chem. Ed. 47, 672-675 (1970)

6. L. Braun, "The Computer Animation Resource Center Workshop", held at 1969 UAIDE Annual Meeting October 19-22,1969 Miami Beach, Florida

7. E.J. Corey and W.T. Wipke, "Computer Assisted Complex Organic Syntheses", Science 166, 178-192 (1969)

8. M.L. Corrin, "Lecture Demonstrations with an Analog Computer", J. Chem. Ed. 43, 579-581 (1966)

9. M.A. Couper, Compt. Rend. Acad. Sci. Paris, 46, 1157 (1858)

10. N.C. Craig, D.D. Sherertz, T.S. Carlton and M.N. Ackermann, "Computer Experiments", J. Chem. Ed. 48, 310-313 (1971)

11. D. T. Cromer, "Stereo Plots of Hydrogen-like Electron Densities", J. Chem. Ed. 45, 626-632 (1968)

12. D. Crowfoot, C.W. Bunn, B.W. Rogers-Low and A. Turner-Jones, "The Chemistry of Penicillin", London (1949)

13. W. Dannhauser, "PVT Behavior in Real Gases: Computer assisted exercises for physical chemistry", J. Chem. Ed. 47, 126-128 (1970)

14. C.S. Ewig, J.T. Gerig and D.O. Harris, "An Interactive On-Line Computing System as an Instructional Aid", J. Chem. Ed. 47, 97-101 (1970)

15. D.E. Geister and R.L. Phillips, "On Line Graphical Data Reduction for Undergraduate Engineering Laboratories", in Proc. Conference on Computers in the Undergraduate Curricula, 3.58-3.64 (1970)

16. J.W. Gibbs, "Graphical Methods in the Thermodynamics of Fluids", Trans. Conn. Acad., Vol. II, 309-342 (1873)

17. R. Glaser, W.W. Rampage and J.I. Lipson, "The Interface between Student and Subject Matter", Learning Research and Development Center, University of Pittsburgh, Pittsburgh, Pa. (1964) p. 177

18. R. Griswold and J.F. Haugh, "Analog Computer Simulation: An Experiment in Chemical Kinetics", J. Chem. Ed. 45, 576-580 (1968)

19. E.J. Hartung, "The Screen Projection of Chemical Experiments", Melbourne University Press, Carlton, Victoria, Australia (1953)

20. Jabir ibn Hayyan, "Summa Perfectionis", Berne (1545) and Brunschwick, "Buch zu Distillieren", Johann Gruninger, Strassburg (1519)

21. W.H. Huggins and D.R.Entwisle, "Computer Animation for the Academic Community", AFIPS Spring Joint Computer Conference, 623-628 (1968)

22. D.A. Humphreys and R.H. Tomlinson, "Closed Circuit Television in Freshman Chemistry Laboratories", J. Chem. Ed. 46, 618-619 (1969)

23. A. Hurwitz, J. Citron and J. Yeaton, "GRAF: Graphic Additions to FORTRAN", AFIPS Spring Joint Computer Conference, 553-558 (1967)

24. E.Hyde and L. Thomson, "Structure Display", J. Chem. Doc. 8, 138-146 (1968)

25. J.D. Joyce and M.J. Cianciolo, "Reactive Displays: improving man-machine graphical communication", AFIPS Fall Joint Computer Conference, 713-721 (1967)

26. K.C. Knowlton, "Computer Produced Movies", Science 150, 1116-1120 (1965)

27. R.S. Ledley and F.H. Riddle, "Chromosome Analysis by Computer, Scientific American, 40-46 (1966)

28. C. Levinthal,"Molecular model-building by computer", Scientific American 214, 42-52 (June 1966)

29. C. Levinthal, C.D. Barry, S.A. Ward, and M. Zwick, "Computer Graphics in Macromolecular Chemistry", in D. Secret and J. Nievergelt (Editors), "Emerging Concepts in Computer Graphics", W.A. Benjamin, Inc., New York (1968)

30. L.A. McGrew, "Student Produced Instructional Films", J. Chem. Ed. 47, 763 (1970)

31. D.M. McEachern, Jr. and P.A. Lehmann, "Calculating Dipole Moments and Atom Coordinates Using Molecular Models", J. Chem. Ed. 47, 389-390 (1970)

32. E.F. Meyer,Jr., "TV Displays 3-D Structures", C & E News, pp. 25-26, May 17, 1971

33. T. Miura, J. Tsuda and J. Iwata, "An application of hybrid curve generation - cartoon animation by electronic computer", AFIPS Spring Joint Computer Conference 141-148 (1967)

34. E.M. Mortensen and R.J. Penick, "Computer Animation of Molecular Vibrations: Ethane", J. Chem. Ed. 47, 102-104 (1970)

35. K. Nagano and D.E. Metzler, "Machine Computation of Equilibrium Constants and Plotting of Spectra", JACS 89, 2891-2900 (1967)

36. G. V. Nelson, "Instant TV Replay in the Lecture Hall", J. Chem. Ed. 46, 620 (1969)

37. A.M. Noll,"Computer Generated 3-Dimensional Movies", Computers and Automation, November 1965

38. A.M. Noll, "Computer Animation and the Fourth Dimension", AFIPS Fall Joint Computer Conference, 1279-1284 (1968)

39. K.F. O'Driscoll, "Patterns in Terpolymerization", J. Polymer Sci.: Part C, 23, 47-58 (1968)

40. Y. Okaya, "Interactive aspects of crystal structure analysis", IBM Systems Journal 7, 322-330 (1968) 227-228 (1970)

41. D.L. Paterson and M.E. Fuller, "Physical Chemistry Students Discover the Computer", J. Chem. Ed. 48, 314-316 (1971)

42. M.F. Perutz, "The Hemoglobin Molecule", Scientific American, 64-76 (1964)

43. Q.R. Petersen, "Some Reflections on the Use and Abuse of Molecular Models", J. Chem. Ed. 47, 24-29 (1970)

44. R.E. Rondeau and H.A. Rush III, "On-line Printing of NMR Spectra", J. Chem. Ed. 47, 139-144 (1970)

45. J.L. Schwartz and E. F. Taylor, "Computer Displays in the Teaching of Physics", AFIPS Fall Joint Computer Conference, 1285-1292 (1968)

46. R.H. Schwendeman, "Computer Simulation of Experimental Data", J. Chem. Ed. 45, 665-668 (1968)

47. L.C. Snyder and S. Meiboom, "Molecular Structure of Cyclopropane from its Proton NMR in a Nematic Solvent", J. Chem. Phys. 47, 1480-1487 (1967)

48. I.E. Sutherland, "Sketchpad", Technical Report #296, Lincoln Labs MIT, January 1963

49. F.D. Tabbutt, "The use of Analog Computers in Teaching Chemistry", J. Chem. Ed. 44, 64-69 (1967)

50. A.C. Wahl and U. Blukis, "Educational Film Loops on Atomic and Molecular Structure", J. Chem. Ed. 45, 787-789 (1968)

51. D.D.Weiner and S.E. Anderson, " A computer animated movie language for educational motion pictures", AFIPS Fall Joint Computer Conference,1317-1325(1968)

52. R.A. Weiss, "BE-VISION, a package of IBM 7094 FORTRAN programs to draw orthographic views of combinations of plane and quadic surfaces"

53. J. Whitney and J. Citron, "CAMP - Computer Aided Movie Production", AFIPS Fall Joint Computer Conference, 1299-1309 (1968)

54. R.L. Wiginton, "Abstracting Chemical Abstracts Service Information", in Proc. IBM Scientific Computing Symposium on Computers in Chemistry, Thomas J. Watson Research Center, Yorktown Heights, New York, October 7-9, 1968

55. K.R. Wilson, C. Parr and J.C. Polanyi, "Computers help chemists produce movies", C & E News, p. 46 November 9, 1970

56. W.T. Wipke, "System uses sound to map molecular models", C & E News, p. 36, April 12, 1971

57. E.E. Zajac, "Computer Animation: a new scientific and educational tool", SMPTE 74, 1006- 1008 (1965)

Communications, Networks and Program Exchanges

Introductory Remarks by the Chairman

David O. Harris

During the decade of the 1960's, the primary emphasis in computing from the chemists' point of view was toward the development and utilization of computing technology for the solution of chemical problems. As a result of this effort, hardware and software systems have become available which are appropriate to many areas of chemical research and education. The cost of the development of these systems both in terms of time as well as dollars has been high and therefore it is important that the results of this work be made available to as large a community as possible. In many cases it is only necessary to provide a prospective user with a card deck or tape which he can run on his own computer system while in others it may be necessary to provide considerable assistance at a more basic level. There are, however, other more fundamental problems. I'm sure that many if not all of the participants at this conference have had the experience of hearing described applications programs, hardware and software systems which could be of value in either your research or educational activities but have found on closer inspection that it is neither possible nor economic to use them at your particular faciltiy.

In the early days of computing it might have been reasonable for each of us to continually be engaged in "reinventing the computer wheel"; however, if we are to take the fullest advantage of our computer tools we must in the 1970's address ourselves to the problem of resource sharing in its broadest context. It is to this problem that this session is devoted.

QUANTUM CHEMISTRY PROGRAM EXCHANGE

and
PROSPECT FOR A NATIONAL QUANTUM CHEMISTRY LABORATORY

Harrison Shull

QCPE was born at the first Gordon Conference in Theoretical Chemistry held at Tilton, New Hampshire, July 9-13, 1962, at which time Professor Shull offered the facilities of Indiana University in encouraging the exchange of computer programs. The exchange was begun by issuing a mimeographed statement of purposes and details, and a questionnaire eliciting interest. The first newsletter with its characteristic Ψ monogram, designed by Dr. Keith Howell as an adaption of the monogram for Indiana University utilizing small Ψ's for an appropriate Quantum Chemistry Symbol, appeared with a dateline of April, 1963. This original issue contained 11 programs, all from Indiana University and a newsletter distribution list of 72 names supplemented by four others in the newsletter itself.

As the exchange grew in size, much of the processing had to be mechanized. Initial programs for accomplishing this were written by Dr. Franklin Prosser, and these have subsequently been modified and added to Mr. Richard Counts. The Air Force Office of Scientific Research provided support for the exchange beginning in the fall of 1964, and newsletter 8 of January, 1965, was the first to bear the cream and crimson covers advertising this generous support. In July, 1964, we began publication of summaries of current research in progress. October, 1964, ended Dr. Howell's association with QCPE when he returned to Great Britain. At that time, Dr. Prosser assumed direct supervision of the exchange.

In October, 1966, distribution of most programs was shifted from punched cards to magnetic tapes. In January, 1967, Mr. Richard Counts assumed administration of QCPE, which he still undertakes. In January, 1968, we introduced handling charges to compensate partially for program distribution costs. The newsletter in February, 1970, brought some inkling that the funding pattern might be expected to change, but it wasn't until April that we received word that support from the Air Force would cease as of June 30, 1970. By stretching funds and by orderly closing-out of accounts, we financed the August distribution as the final Air Force report.

A final report on previous finances is in order. Through June 30, 1970, a grand total of $108,651 was received from the AF

OSR in support of QCPE for a six-year period. In addition, Indiana University contributed approximately $16,600 in computing time and in released time of project supervisors, not counting any of Professor Shull's time. As direct receipts from the membership through June. 1970, largely in payment of postage costs for programs mailed out, we received $2,200. It is a pleasure to acknowledge the enthusiastic support of Amos Horney of the Air Force Office of Scientific Research and of his colleagues during this period. We know they regret as much as we do the termination of their support.

Negotiations were at once entered into with the National Science Foundation Office of Computing Activities to provide interim continuing support of QCPE. Beginning in October, 1970, partial support for a three-year period was received covering full anticipated expenses for 1970-1971, two-thirds of expenses anticipated for 1971-72, and one-third for 1972-73. In the meantime, it is hoped that membership fees and program distribution fees will build up to permit continuation of the exchange.

As of June 3, 1971, our active membership role was 826, down from about 1,500 free memberships the previous August. For this first year of operation we budgeted fee receipts at about $15,000, and through June we had received almost $11,000, only slightly below our expectations. Except for the period from July 1 to October 15 when we had zero support, we would be approximately on schedule in developing a self-supporting activity by the end of the third year. The crucial question will be, however, to what extent memberships are renewed and active program contributions and use continue. It is our hope to maintain QCPE as an active live organization for transmission to a National Quantum Chemistry Laboratory if the latter can be established.

The report of the Bethesda Conference leading to a recommendation of that group for formation of such a Laboratory has been distributed to the participants. The report was reviewed by the Committee on Science and Public Policy of the National Academy of Sciences, which quite properly declined to lend its backing at this stage to the positive recommendation for such a Laboratory. COSPUP did, however, support a study in depth by the chemistry community of the need for and feasibility of such an organization, and the Governing Board of the National Research Council has agreed to sponsor such a study if funds can be found. Negotiations are currently in progress with the National Science Foundation to mount such a study.

We want to make it very clear that the proposed Laboratory is not designed primarily as a research tool for Quantum Chemists, but rather as a computational facility capable of the production of theoretical computations of use to the general chemistry community for the solution of chemical problems. The committee to study the Laboratory if funding is obtained, will therefore be broadly based, representative of the total spread of chemistry, and in addition will include experts knowledgeable in computer and communication technology.

We append here a table indicating the growth in membership and use of QCPE during these years. We wish to point out that the membership figures represent active members only. The list has been culled each year to eliminate the names of those who no longer maintain an active interest in QCPE.

Date		Programs in Library	Number of Participants	Programs Sent Out (cumulative)
April	1963	11	72	0
January	1964	30	135	40
January	1965	57	293	450
July	1966	88	430	860
January	1967	104	576	1,495
January	1968	115	759	3,300
January	1969	140	~1,000	6,600
January	1970	153	~1,400	10,300
August	1970	170	1,500	~ 12,000
June	1971	192	872	13,072

A FORWARD LOOK

Lawrence G. Roberts
Advanced Research Projects Agency

Computer-to-computer data communications today might be compared with people-to-people communication via telegraph before the day of the telephone. Sending a message by telegraph was so slow that the media could only be used for non-interactive transmission of essential information. As such, its use was limited. The telephone provided an ability for people to interact, thus permitting a whole new range of applications. Considering people somewhat mechanistically, we might view their use of the telephone as inter-human resources sharing. To solve a problem a man will call those people who have bits of data which he needs and will call on specialists for opinions, thus making use of other human resources. This is achieved because the media is appropriately responsive for human requirements and permits interactive conversation, thus eliminating the need for transmitting excessive detail, much of which may be unnecessary. Also, with an interactive dialog, information does not need to be formatted in a standard way since details can always be clarified if misunderstood. This increase in utility and the many new applications thereby permitted have resulted, as we know, in a vast increase in telephone traffic volume over telegraph traffic levels.

Communication between computers would most likely be effected in an analogeous manner if a data communication system were made available which matched the needs of computers as well as the phone matches the needs of humans. Such a system would, of course, have to have different technical parameters such as connection time, data rates and reliability, than those required for voice communication, but if it permitted truly interactive conversations between a large ensemble of computers, the effect should be much the same in that it would permit remote access to specialized hardware and software resources, joint problem solving and the dynamic retrieval of data from remote files. The analogy with the telephone is just one way of examining the potential impact of substantially improved data communication between computers and the resultant increase in applications and traffic which such a change might bring about.

Intercomputer communication has many quite substantial differences from interpersonal voice communication. Whereas voice conversation is a rather continuous, constant data rate process, communication with computers, either from computer consoles or other computers, requires a burst transmission rate several orders of magnitude higher than the average rate, even during a single conversation. Since there has been very little experience so far with real intercomputer traffic where two programs are talking to each other, it is useful to examine the characteristics of computer console traffic which is both a component of, and is also likely to have the same general parameters as, computer-to-computer traffic. From statistics on teletypes, graphic consoles and remote batch stations, it appears that the ratio of burst rate to average rate is approximately 100 to 1. This means that if a standard communications line is established for a computer conversation, the average utilization of that line will only be about 1% and therefore the cost will be 10-100 times higher than the raw cost of moving the bits. A second characteristic of computer-to-computer communications is that the connect time to establish a conversation must be short enough that the computers or the computer users are not held up unduly when the need to access a special resource is determined. For computers the "connect time" should be considerably less than a second as opposed to the 20 or 30 seconds commonly experiences for voice communications. Third, the maximum data rate required in man-machine interaction must be considered. It is known that for useful comprehension by a human, the peak data rate for graphical material is on the order of 20 Kilobits per second, which suggests the required bandwidth for console-to-computer communications. This also suggests at least a minimum for computer-to-computer communications. Finally, the error rate for intercomputer traffic must be far lower than required for voice communications or computer console traffic since there is usually very little, if any, redundancy inherent in the data. For many applications the error rate must be less than one in 10^{12} bits. At the same time, the reliability (up time) of the data communications system must be very high if the user is to depend on remote resources. The cost of a data service providing the characteristics outlined above must be compared with the cost of duplicating the computer resources involved. Very simply, if the monthly cost of adequate communications service exceeds or even approaches the cost of a reasonably well endowed computer installation, it is not likely to be economical to use that communications service rather than duplicate that facility. Arbitrarily setting a threshold at 20% of a computer facility cost, it can be predicted that the communications system should not cost more than $10K per month per node.

A few years ago no communications system in existence even came close to providing the type of service just described. Therefore, ARPA undertook to develop such a capability so as to make resource sharing between computers possible. The communications system which resulted is utilized in the ARPANET and currently interconnects more than 20 computers at 15 locations around the country. By early 1972, expansion to 25 locations is expected (Figure 1). A delay-engineered message switching system, the ARPANET consists of Interface Message Processors (IMPs) at each node intercommunicating over 50 kilobit leased communication lines and connected to one or more Host computers at each site. The IMP accepts messages from the Host, breaks them into thousand bit packets and sends each packet toward the destination over whichever communication line is currently optimal. Each IMP in turn checks the error detection code on the packet and, if it checks, routes the packet on to the next node and sends an acknowledgment to the previous node. At the destination, packets are assembled back into a message and delivered to the Host. In practice this organization proves to be extremely responsive, delivering short messages anywhere in the country within .1 second and permitting throughout rates for long messages of up to 80 kilobits per second. By adjusting the number of communication lines which are leased, the network can be engineered to have almost any desired overall average capacity between 2 kilobits per node and 60 kilobits per node. Since each communication line is being used for traffic between many pairs of nodes simultaneously, it can be loaded quite efficiently even though the individual Host-to-Host conversations have such a high ratio of burst rate to average rate. The actual cost of the total network communications system including the cost of IMPs, maintenance and communication lines ranges from $3K to $6K per month per node, depending on the overall traffic levels and the facilities required at each node. For new people entering the network, the February, 1972, network of 23 nodes is currently estimated to cost $4.8K/node/month; $3.1K for an equal share of the communication lines cost and $1.7K for the lease of a minimal IMP. If a user wishes to provide direct console access to the network, a Terminal Interface Processor (TIP) would be used. The TIP, which will become available in August, 1971, will act both as an IMP and as a simple host, permitting up to 64 consoles and peripheral devices to intercommunicate with any host in the network at rates up to 20 kilobits/sec. Thus the TIP expands the network concept to include nodes without an interactive host of their own, but who wish high bandwidth support for graphic consoles, printers, and large collections of lower speed devices. Use of a TIP increases the cost by $1.6K/mo.

Although an equal share of the communications line cost is currently allocated to each node, this policy will be changed, as soon as feasible, to one of charging only for the bits actually sent from each node. Referring to Figure 2 it can be seen that the cost of the network increases almost linearly with capacity, at least for bandwidths below 16 KB/node. Also, it turns out that the capacity and cost of these distributed networks are remarkably insensitive to the distribution and destination of traffic, the total traffic being the only important parameter. Thus, it is appropriate to charge for traffic initiated at a node, based on the cost of increasing the total capacity of the network by that amount. From Figure 2 it can be seen that this will be 11¢/megabit for the ARPANET. However, since the network cannot be expected to be always fully loaded to peak capacity, day and night, it is likely the actual rate will be 30¢/megabit based on an estimated 36% average loading. The total cost per node would then be $1.7K/month plus 30¢/megabit.

Looking ahead, assuming the broad availability of a data communications service similar to the ARPANET system, it is clear that very significant changes in computer system organization will take place. Some of these changes will occur rapidly--within the first five years--and others will take a decade or more before people fully accept the concepts. Soon after a network with a dozen or more reliable computer services becomes available, many institutions will find it far more economic to obtain their computing services from a selected set of these remote systems, rather than run their own computer center. For example, take the case of an institution about ready to upgrade its facility. One choice would be to obtain a medium scale, general purpose batch system, admittedly a compromise for their large numerical users and time-sharing users, but the best single system that they can afford. Alternatively, they could buy no new machine and obtain access to several of the systems on the network through a Terminal Interface Processor. This approach permits their large numerical users to use a large "number-cruncher", their statistical and payroll users to access a large scale general purpose system, and their interactive users to have teletype or graphic console access to a good time-sharing system. Overall the cost of each service is less than it would have been on a dedicated G-P computer by factors between two and ten. Also, they can buy just the capacity they need and expand smoothly rather than having to pay for an oversize machine for a year or two. The peripherals cost the same in either case and the network cost is negligible compared to the direct computer cost savings. As added benefits, the computer services they use are probably better run and more reliable than they could hope to do themselves since the services must stay competitive; a wider range of software is available and can be accessed directly without translation or transfer; and as new hardware is introduced which is economically useful, they can transfer jobs to it on a selected and liesurely basis.

The direct use of distributed hardware services just described will probably account for most of the initial use of the network. This growth should proceed about uniformly over the next eight years--two computer replacement cycles. Some additional traffic will be introduced by the gradual transfer of current data traffic from other data communication networks to the computer network due to the economy or reliability, but the total quantity of this traffic is minor in comparison to the new traffic generated by the computer resource sharing activity.

A second major application of the computer network is data base sharing--direct retrieval from remote, one of a kind data bases. Currently, when large data bases or files are needed at several computer centers, duplicates are maintained at each center. This difficult and costly practice can be avoided if the access speed through the network is fast enough so that neither human users nor computer processes are unduly delayed. The ARPANET response speed of one-tenth of a second for a question and three-tenths of a second for a one-page answer is quite acceptable for a human user and for a computer program it is no worse than a slow disc. To start with, this response appears adequate; however, further experience may indicate a need for faster response in future networks. Data base sharing will not build up as rapidly as hardware service sharing,however, since it represents only an incremental saving for an installation and demands considerable faith in the network. Copying a 10^9 bit data base monthly might cost $2,000--less than the minimum network cost and therefore not a prime motivation for joining the network. However, the cost of accessing the data base through the network would cost at most $300 even if all the data were required, providing a considerable cost saving and convenience as long as the network connection had other justification. Of course for very large data bases such as the 10^{11} bit weather-climate data base being developed by the Air Weather Service for the ARPANET, the cost of either copying or storing a duplicate would immediately justify network connection.

In most cases very large data bases would not be developed at all without a network making possible nation-wide access, since the cost would be prohibitive. Data base sharing, therefore, is not likely to grow rapidly until the network is reasonably well established, lagging the service-sharing growth by perhaps two years, but then growing exponentially as everyone requests access to all the information available.

Software sharing, the third major application, is the remote use of software subroutines and packages, programs not available on the users' primary computer due to imcompatilibity of hardware or languages. An example of this type of activity might be the use by MIT scientist of the Stanford Heuristic Dendral System, a program for determing molecular structure given the mass spectrum. On a computer at MIT the scientists would collect and preprocess the mass spectrum data. Then, much like using a subroutine, the Stanford computer would be called, the data sent and the molecular structure, when determined, sent back. If interaction were required, the MIT scientist would be interrogated much as if he were at Stanford, thus building up the heuristic model based on nation-wide inputs. The MIT computer, upon receiving the response, would proceed locally with the calculations or displays desired. Software sharing like this will be required if we are to maintain maximum progress as the volume of useful software continues to expand. Since the annual cost for software is already larger than that for hardware and, to some extent, should be cumulative rather than wearing out, the long range importance of software sharing is clearly greater. However, due to human inertia and a strong "not invented here" syndrone associated with software, it is clear that the cross-utilization of software will take years to develop. The buildup of software sharing activity will most likely begin very slowly, growing exponentially, but not become a major factor until the network becomes well established in four to eight years.

Besides hardware, software, and data base sharing, there are many other important network applications, all of which require a large viable network before they become important in their own right. These include teleconferencing, publishing, library services and office paperwork filing and distribution. Ten to twenty years from now these applications may well dominate computer usage and network usage but they are not likely to be important factors for at least five years.

Overall, then, hardware service sharing is likely to be the major factor causing networks to come into existence since the effective cost of computing can be drastically lowered for only a moderate communications cost. Then, data base sharing will become the dominant force expanding the traffic in three to four years. Software sharing, although very important in the long run, will not become a major factor for four to eight years. The text oriented services, libraries and office work, will then come into their own in ten to twenty years. The whole trend should decrease the importance of the general purpose computer as stand-alone systems, and substantially increase the importance of specialized systems--ones which can provide a specific service at the lowest cost.

ARPANET
MARCH 1972

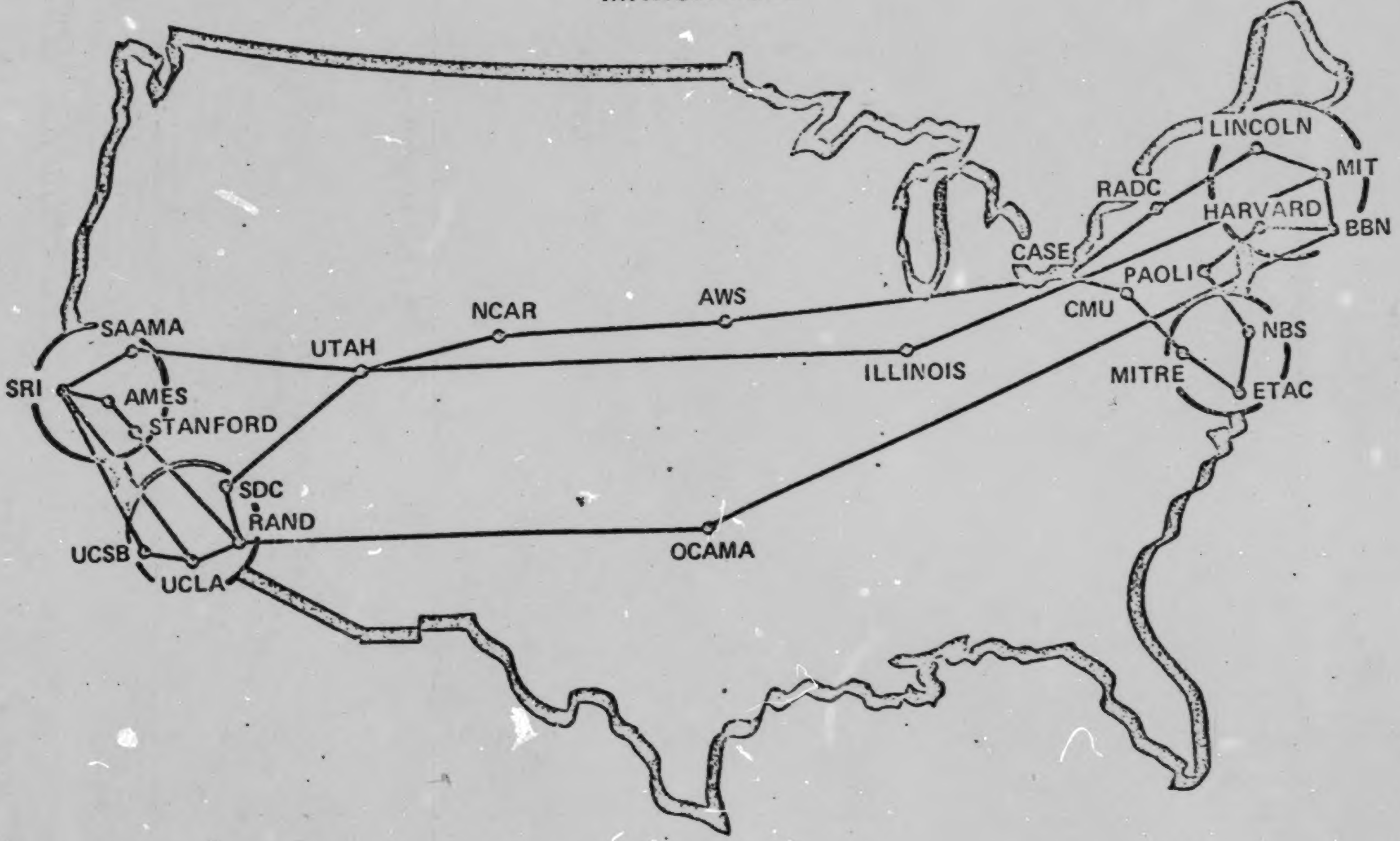

SRI
SAAMA
AMES
STANFORD
UCSB
UCLA
SDC
RAND
UTAH
NCAR
AWS
ILLINOIS
OCAMA
CASE
CMU
RADC
LINCOLN
MIT
HARVARD
BBN
PAOLI
NBS
MITRE
ETAC

FIGURE 1

ARPA NETWORK-COST VS CAPACITY

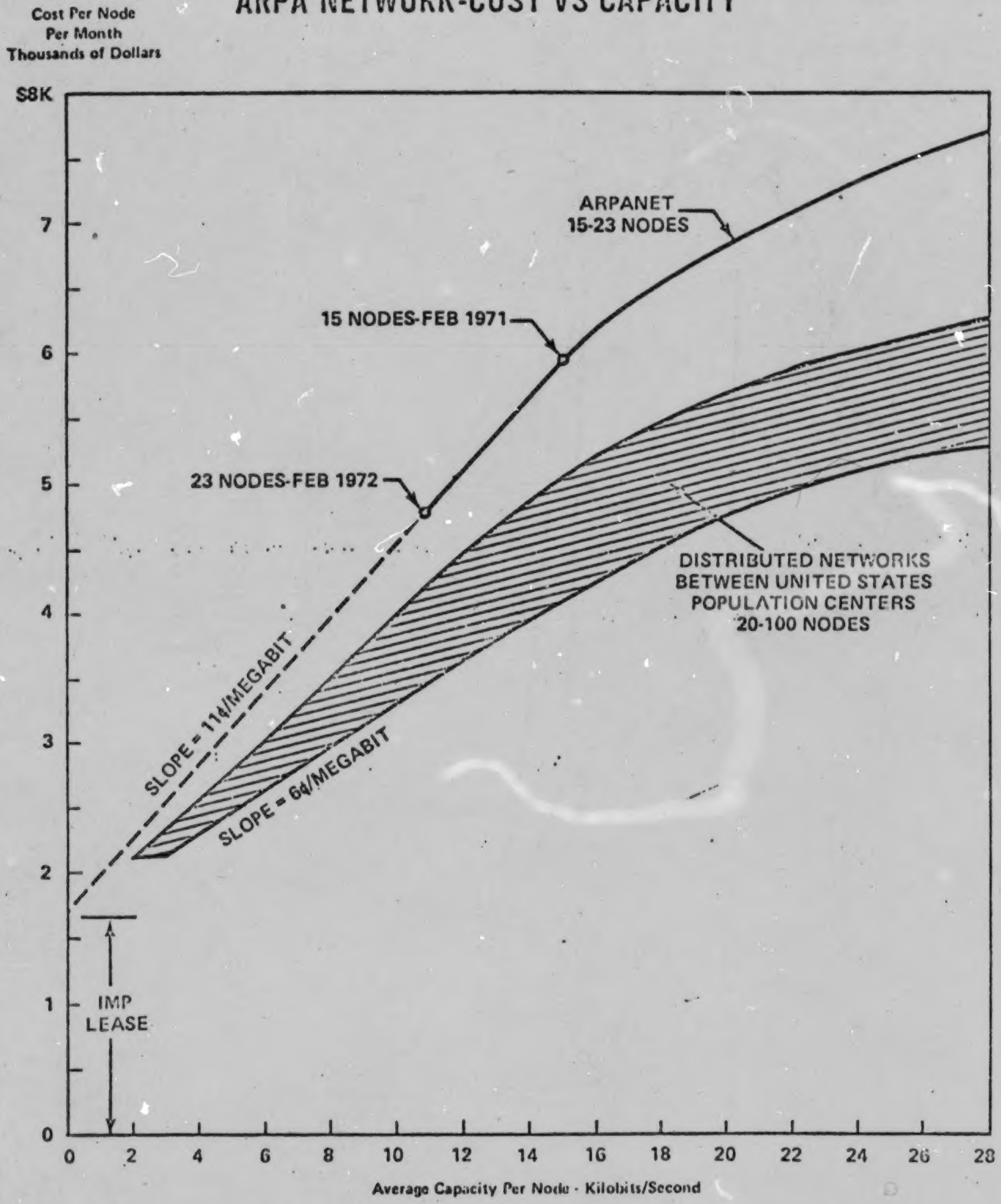


FIGURE 2

Regional Computer Networks and Curriculum Development

Peter G. Lykos

Illinois Institute of Technology*

Introduction

1971 is the 25th anniversary of the first large-scale electronic computer, ENIAC, designed and built by Professors Eckert and Mauchly at the University of Pennsylvania.

Today some 100,000 computers are in operation around the world, 70,000 of them in the USA.

A widespread unit record punched card information storage and handling technology constituted a fertile environment for the computer which led to its rapid proliferation, operating in stand-alone mode, soon after its invention. More recently the computer has come to be interfaced with, and used as an important component of, our communication network. A second rapid growth period in computer use is developing with a corresponding rapid increase in number and variety of terminal devices providing more convenient and flexible access to computer utilities.

The computer, more accurately named the information processing machine, has impacted every area of human endeavor. Yet its potential in the service of mankind remains largely untapped; its principal applications continue to be only the two conceptually primitive ones: as a giant desk calculator serving engineers and scientists, and as a giant accounting machine serving administration.

Computers in Higher Education

Computers came to a few college, and to many university campuses in support of administration and in support of graduate research, primarily as a giant desk calculator.

At those campuses with extensive research activity, the computation centers grew very rapidly, depending very heavily on federal support, and having as their primary objective support of graduate research.

Relatively little use was made of the computer in undergraduate programs, let alone in high school curricula, even though it was becoming evident that the problem solvers and the decision makers of our society were discovering new and more powerful techniques for doing their work which depended in an essential way on the computer. Clearly Education was failing to meet its responsibility of preparing the student of today to be the citizen of tomorrow.

---

* On leave to Office of Computing Activities, National Science Foundation

NSF Regional Networks

In September of 1967, the newly formed Office of Computing Activities of the National Science Foundation convened 30 Computer Center Directors from around the country in order to discuss the feasibility of diffusing their computer expertise and computer service to neighboring institutions of higher learning, particularly colleges. Of those 30, nine submitted proposals which were funded in FY 1968 and thus were created the first regional networks, namely:

St. Anselm's College, New Hampshire, plus five colleges
Carnegie-Mellon University, Pennsylvania, plus ten colleges
Purdue University, Indiana, plus six colleges
University of Iowa, Iowa, plus ten colleges
Southern Regional Education Board, Georgia, plus ten colleges
Illinois Institute of Technology, Illinois, plus nine colleges
Cornell University, New York, plus five colleges
Dartmouth College, New Hampshire, plus nine colleges
Oregon State University, Oregon, plus six colleges

Since that time the number has grown to 23, supporting 212 colleges (10% of the nation's total universities and colleges), with the emphasis shifted from computer programming instruction and computer service to discipline-oriented faculty training and curriculum development.

Funding and Future of University Computer Centers

The de-emphasis on funding computer service per se has corresponded to a fairly abrupt and sharp reduction in federal support of university computer centers in general. That, together with the general depression of the economy over the past two years, has created a major support problem for university computer centers approaching a crisis situation.

Additionally university research scholars whose research progress has as one limit the size and speed of the computer available to them, are working at a level below that of current technology because no university has on its campus the most powerful computer extant.

A recent two-year study by the RAND Corporation suggests that the day of the single large-scale university computation center, attempting to support the entire range of computational needs of administration, research, and education, is fast passing. A multifaceted commercial computer utility spectrum is evolving, made all the more viable with the increase in the data communication sophistication of the communication industry.

That study also suggests that development and wide distribution of computer-based curricular elements will occur at such time as the commercial sector, such as the textbook publishing industry, enters the picture with a royalty system for the authors and a standardization and distribution system for the audience.

Neither of the two developments foreseen by the RAND study yet has manifested itself, however, and the NSF developed system of regional networks, which has caused a rapid expansion of undergraduate computer use, both in amount and in discipline range, continues to be the major force. Not the least of the factors operating is the low cost possible. For example, using IIT's Remote Job Entry fast batch system with turnaround times of less than 5 minutes, contact time charges for TTY's of $2.00 per hour, and an efficient compiler with minimum charge of 25¢ per run, has permitted as many as 12 small student programs to be batched together as one run before the minimum charge for UNIVAC 1108 CPU and channel time of 25¢ was exceeded.

Curriculum Development

The three-day NSF-supported Iowa conference of June 1970, designed to deal with broad brush use of the computer in undergraduate curricula, drew 740 participants and hundreds of papers. Its successor, the Dartmouth conference of June 1971, drew 1200 participants and hundreds of papers.

Most of the creative effort has been limited to producing computer programs for handling specific classes of problems where the computer is used as a desk calculator. However simulation, particularly in economics and management, and non-numerical applications, are beginning to appear. Also use of the small dedicated computer in a control environment is represented in at least a dozen colleges and universities in the undergraduate program. These specific applications have not yet caused any sweeping revision of courses, let alone of curricula. That is yet to come.

As a distribution medium, the standard journals serving the several disciplines are still groping trying to discover whether algorithms are publishable and, if so, in what form.

Conferences with proceedings constitute yet another distribution medium.

Computer program exchanges are a viable means of distributing computer programs and corresponding descriptions. These have sprung into being as part of several of the regional computing activities and, at Eastern Michigan University, as a regional, discipline-oriented entity which exists as a collection and distribution center, and as an education center. There has not yet appeared a comprehensive, nation-wide, standardized program exchange such as the research-oriented QCPE.*

---

* Quantum Chemistry Program Exchange, Indiana University

State-Wide Networks

The trend seems to be toward regional networks organized on a state-wide basis. This has already happened in North Carolina and in Georgia on a large scale, and many other states are well on their way to similar systems. Such systems are involving both the public and private institutions although they tend to be dominated by the public system. Exceptions occur in that multistate cooperatives are being formed involving low population states and in that metropolitan cooperatives are being formed in major cities.

The IIT Regional Network--A Particular Experience

One of the original nine NSF-supported regional networks, the IIT network, was unique in that it had at the onset faculty training and curriculum development as its primary thrust. That project embodies all the elements of a regional network, even those recently formed, and the Introduction and Summary from the final report for the project is herein appended in order to impart some of the flavor of regional networking and to serve as a concluding statement for this paper.

FINAL REPORT ON THE IIT-CENTERED REGIONAL COMPUTER FACILITY, FACULTY TRAINING, AND COOPERATIVE PROGRAM IN CURRICULUM DEVELOPMENT: INTRODUCTION AND SUMMARY

PETER G. LYKOS

History and Objectives

In July, 1968, the National Science Foundation made a grant to Illinois Institute of Technology and nine participating colleges and universities in northern Illinois and southern Wisconsin for purposes of engaging in a joint program in the development of a regional computer network and in faculty training and curriculum development involving the introduction of the computer as an instructional tool in undergraduate courses. The project was for a combined total of $885,000 for a period of 26 months ending September 1970 (subsequently extended through June 1971). In addition to Illinois Institute of Technology, the nine participating institutions were:

Chicago City College (four of eight campuses)
Dominican College (Wisconsin)
Elmhurst College
Loyola University (two campuses)
Monmouth College
Mundelein College
Quincy College
Ripon College (Wisconsin)
St. Xavier College

In 1969, Blackburn College and Wheaton College joined the project on a self-funding basis.

The main group (9) of participating institutions consisted of liberal arts colleges. The exceptions were a large university (Loyola), a large city college system (Chicago City College, with nine branches), and an institute of technology (IIT) specializing in science and technology.

Except for IIT, none of the participating institutions had stand-alone computer facilities. In fact, all had not previously used computer facilities to any significant extent. In those cases in which there was previous experience, it was initial and preliminary and the same applied for instruction in programming.

IIT, as the leader institution in the setting up of the regional computer network, had a modest medium-scale computer facility, an IBM 360/40 with an IIT-developed remote job entry system and a conversational language supporting remote teletypewriters (later changed to a UNIVAC 1108 configuration). In addition, IIT had extensive experience over many years in programming language instruction, ranging over diverse groups, from high school students through college instructors.

The essential objectives of the project were:

1. Development of operating remote terminal facilities at the other participating institutions accessing a common flexible computer.
2. Stimulating programming language instruction and other means of developing a user community of computer facilities at the other participating institutions, and
3. Faculty training and faculty involvement in curriculum development involving the use of the computer as an instructional tool in specific discipline course work.

Since IIT had been operating a highly successful computer network for remote users for some time, the basis of the mechanics of a regional computer network was already present and there was little wasted effort in extending the existing network to these other institutions (although the radical change in central computer system midway through the project worked a major hardship on the remote terminal users). There were then no major difficulties in making arrangements for each participating institution to have operational remote terminal facilities relatively quickly.

The second of the objectives listed above was also relatively quickly accomplished. In this respect the previous extensive IIT experience and the limited experience of some of the groups served as examples of operating mechanisms of instruction.

The major thrust of the program thus was the third objective listed above. This was one stage of "nondevelopment" which all participating institutions had in common. In essence the primary objective of the program was the unique area in which there was almost a complete lack of previous development, or even planning for development thereof. It is most interesting, and indicative of the very real need for the support of a program such as this, that, even at IIT with its extensive computer facilities and its extensive background in instruction of this project, no formal or even informal activity in the area of the development of curriculum materials which make use of the computer as an instructional tool existed.

Project Structure

Each participating school recognized that faculty training and faculty participation in curriculum development was the most important objective. Each participating campus identified a number of faculty in a variety of disciplines for participation.

The basic operational mechanism used was to organize the participants into discipline study groups. Each study group had a group leader drawn from one of the universities in the greater Chicago area. These groups were:

Biology
Chemistry
Physics
Mathematics
Business and Economics
Psychology and Education
Sociology

Conspicuous by its absence is Computer Science which was highly developed at IIT but nonexistent at the other participating institutions except for a data processing program in one of the non-participating Chicago City College campuses.

During the first year, Psychology, Education, and Sociology were lumped into one study group. At the end of the first year, Sociology was made a separate group while Psychology and Education were left together.

Group Leaders' and Administrative Reports

At the end of the project, reports were received from each of the seven group leaders and from the Project Manager (who managed to acquire a Ph.D. in Marketing at Northwestern University during the project period). These reports were edited by the Project Director and the Project Manager and constitute the last section of this report. Each contains an account of what happened in each discipline together with observations and helpful hints, some of which transcend the particular discipline. Accordingly, regardless of the reader's discipline orientation, all the accounts will contain something of interest.

Intermediate project reports produced earlier in the project have been collected as Appendices and they contain much useful information regarding project structure, project start-up, bibliographies, program exchange listings, financial analysis, etc.

A brief analysis of the computer in higher education was made by the Project Director (Attachment 1 to this Introduction and Summary) presented as one of twenty "provocative" statements at the RAND Invitational Conference on "Computers in Instruction: Their Future for Higher Education" held in Beverly Hills, California, on October 1, 2, 3, 1970. That analysis is based in part on the many experiences of the Project Director both in organizing and in conducting the three-year project.

Project and Project-Inspired Conferences

In addition to the several "in-house" conferences on curriculum development which included many visitors and participants from outside the project, two additional conferences of national scope were a direct consequence of this project.

The first such was the week-long conference "Computers in Undergraduate Science Education: Physics and Mathematics" held at IIT on August 17-21, 1970. The conference was cosponsored by IIT, The Commission on College Physics, and funded by the Office for Computing Activities of the NSF. The IIT project group leader in physics was codirector of the conference and proceedings were generated. There was considerable emphasis on computer assisted instruction.

The second national conference was the week-long conference "Computers in Chemical Education and Research" held at Northern Illinois University on July 19-23, 1971. The conference was cosponsored by NIU, American Chemical Society, National Research Council, and funded by the Chemistry Section and Office of Computing Activities of the NSF. The Project Director organized the conference and proceedings were generated. The theme of the conference emphasized the fact that the computer is affecting how chemists do chemistry, correspondingly, there need to be established channels of communication between those chemists at the frontiers who are developing innovative approaches to chemical problems tapping the potential of the computer, and all other chemists, particularly those responsible for chemistry curricula. The idea here is that whereas some existing chemical problem-solving methods can be speeded up using the computer, such a very limited approach falls far short of realizing the potential of the information processing machine.

Participating Campuses' Post Project Computing

A few comments are in order regarding what the participating campuses are doing now.

For example, the Chicago City College System has created an IIT-Faculty Training and Curriculum Development-like project at Wright College, one of the four City College campuses which did not participate in the IIT-centered project. Six faculty members from six disciplines have been released part-time from their teaching duties for this purpose and they are being guided by a Wright professor with considerable experience and many contacts in the "Computers in Higher Education" milieu.

Quincy College, which is geographically isolated, has replaced its teletypewriters with a programmable terminal and is using both the University of Iowa IBM 360 and the IIT UNIVAC 1108. This constitutes a conceptual breakthrough in that the faculty are free to shop around having different menus to choose from.

Ripon College has purchased a mini-computer which supports teletypewriters in time-sharing mode. In addition, there is an attempt underway to adapt that system so it can operate as a medium speed programmable terminal to access other computers elsewhere such as at Madison and Milwaukee, Wisconsin, and in the greater Chicago area. Finally, Ripon is creating a small network of its own involving local high schools including faculty training.

The remaining schools have either arranged for their own local mini-computers or are continuing to get service via telephone lines from remote computers.

In every case there has been a significant impact on the academic program in an amazing variety of ways. The principal inhibiting factor operating at all participating campuses continues to be lack of faculty training and interest.

Greater Chicago Area Computer Consortium

The State of Illinois Board of Higher Education recognized the potential of individual institution program-enhancement through cooperation with neighboring institutions and thus stimulated the formation of a committee representing the thirty-odd institutions of higher learning in the greater Chicago area. The Project Director chaired the subcommittee charged with cooperation involving computers. That subcommittee's report, subsequently approved by the Board of Higher Education, outlined a cooperative program evolving through three stages based essentially on the experience derived from this project.

Computational Laboratory

Although the principal problem continues to be training of faculty on the impact of the computer on their discipline content (as well as an awareness of the current state of the embryonic computer-assisted pedagogy), nevertheless, the secondary question of computer support for education has not yet been adequately addressed even within today's standard computer technology. Accordingly, a second outgrowth of the IIT-centered regional network has been the design of the IIT Computational Laboratory (Attachment 2 to this Introduction and Summary) which serves in two modes:

1. Interactive compilers, file system, and a CAI system from a number of CRT terminals with graphics capability as a local computing facility.

2. Programmable medium speed terminal so the instructor can collect student inputs and transmit them to other computers elsewhere for batch processing and subsequently return transmission of results in order to access computer programs no matter where they are running.

A comprehensive overview of available standard hardware has revealed an extremely powerful system (comparable to an IBM 360/40 to 50) at surprising low cost. For example, given the intermediate 16 terminal configuration and a five-year lifetime, a student contact hour at a terminal costs two dollars. (The CRT terminals cost less than keypunches.) However, since the system is designed to make easy and flexible use of computer utilities elsewhere, its useful life in this rapidly changing computer technology actually extends beyond five years.

There exist two systems which are receiving national attention as the basis for a five-year demonstrational project in CAI, namely, the PLATO system based on thousands of non-standard innovative terminals accessing a large computer supporting many academic institutions at all educational levels, and the TICCIT system based on a local stand-alone mini-computer supporting video-compatible terminals designed with the lower educational levels in mind. In part, the IIT Computational Laboratory constitutes the higher educational counterpart to the TICCIT system in apposition to the PLATO system. However, it transcends both systems in that an essential part of its design is convenient access to application computer programs, no matter in what language they were written and no matter on whose computer connected to the country's communication system they are running.

Recommendation

Finally, if one had to distill out of all this experience the single most important recommendation it would be the following:

Innovators in specific disciplines doing research are discovering how man's problem-solving and decision-making power is being extended with the advent of the information processing machine. Strong encouragement needs to be given to them to conduct faculty training institutes and to adapt their discipline-oriented computer-based innovations for use as part of high school, college, and university academic programs.

June 30, 1971.

THE IIT COMPUTATIONAL LABORATORY

Peter G. Lykos
Presently on leave at the Office Computing Activities,
National Science Foundation

Robert B.K. Dewar
Illinois Institute of Technology

AN OVERVIEW

The computer is beginning to find an important place in college education. The influence is not restricted to the hard sciences, but extends throughout the curriculum. However, as recent articles indicate (Educational Media: November 1969, February 1970, September 1970: Proceedings of Conference on Computers in Instruction: Their Future for Higher Education, RAND, October 1,2,3, 1970, Beverly Hills, California). The use of computers has been on only a token basis, even at most of the large universities. This is due to two causes: 1) The faculty have not yet updated themselves and the curricula they teach in order to incorporate computer-based problem-solving techiques: and 2) Even if they had done so, the computer systems currently available would not support broad-brush use of the computer by students in a cost/effective fashion.

It seems essential to incorporate the use of the computer into the curriculum in an orderly scheduled manner. It is quite unsatisfactory to attempt to extend the typical current systems in which students contend for a small number of key-punch machines or terminals in their spare time.

We have therefore devised the notion of a Computational Laboratory, consisting of a classroom of quiet CRT terminals, into which students are scheduled as part of their standard program of study. This laboratory would be used both for computer-based instruction and problem solving sessions in a variety of disciplines.

The availability of low cost mini-computer hardware makes the establishment of such a facility practical at relatively low cost. The mini-computer system can duplicate most functions of a large scale system with respect to the requirements for student computing.

An additional and important degree of freedom is the availability of communications hardware which allows the use of a variety of large scale systems, remote from this installation, to handle more complicated problems. This latter approach is of particular importance, since the possibility of using application programs on the computer system for which they were designed allows the establishment of a direct link between the student in the Computational Laboratory and the frontiers of research in universities across the nation. It should also be noted that the concept of an onsite facility communicating with a variety of large systems, at the convenience of the user, is much more attractive than the more standard arrangement of a small machine which is "tied" to one large central computer.

The necessary software partially exists in a variety of systems. However, no system currently available provides the necessary software in an integrated package designed specifically for this purpose. It is the intention of IIT to generate the required software system, with a view to duplicating the Laboratory in a variety of colleges and universities.

Another important factor with regard to the use of this system in a small college, or a high school, is that it would be capable of supporting the administrative requirements if suitable software were available. Thus funds could be combined to put the system in reach of many small colleges.

No fixed cost limits have been established for individual components. However, a target cost of $200,000 for an intermediate size 16 terminal configuration has been established for the entire system (all items are to be purchased outright). In general, cost is an important factor, since the attraction of this system for small educational units lies in its flexible facilities and low package price. The range of purchase costs is from about $125,000 for an eight terminal system to about $375,000 for a forty-eight terminal system.

It is difficult to assess the total impact of such a system on secondary schools and on higher education. However, the following may be noted:

1. A recent survey by the American Institute for Research of the 23,000public high schools in the USA reveals that 13% are already using the computer as part of the educational process. Presumably a similar percentage applies to the 8,000 private high schools.

2. A recent study by the RAND Corporation of instructional uses of the computer in the 2,600 colleges and universities of the USA reveals that over half make significant use of the computer as part of the instructional process and, furthermore, programmable medium speed terminals (one of the modes of operation of the IIT Computational Laboratory) increasingly will be used to enable college students to access a variety of computer services as a regular part of their academic training.

3. A recent development of Advanced Research Projects Agency has spanned the USA with a medium speed data transmission network dedicated to interfacing computers with each other or with data terminals. EDUCOM is examining the very much underutilized ARPA Net with a view to bringing it into the service of higher education on a national scale.

4. The High-Energy Physics Advisory Panel to the Atomic Energy Commission, Division of Research, recommendation that the CDC 7600 recently installed at the Lawrence Radiation Laboratory be operated as a national facility, has been accepted and scientific users outside of the high-energy physics community will also be accommodated on a time available basis. The system is accessible on a Remote Batch Service basis with large on-site storage facilities available. As of 23 April 71, federal agencies and cost-type government contractors are recharged at the internal rate of $155 per hour for 6600 time. Rates for the 7600 system are not yet determined, but will be adjusted so that costs for equivalent jobs will be less on the 7600 than on the 6600.

OUTLINE OF THE PLANS

Introduction

The IIT Computational Laboratory is an integrated hardware/software package with local and remote processing capability which is intended to meet a broad spectrum of educational needs at the college, as well as the university, and to an extent, the secondary school levels.

The design is based around a carefully selected mini-computer system and derives from the extensive experience of IIT in the field of educational use of computers. Among the important considerations are the following:

(a) It is necessary, for psychological and other reasons, to have local computing capability.

(b) A wealth of applications programs is currently accessible via telephone lines, given a system which can communicate with a variety of large machines.

(c) Computing in the educational environment needs to be a scheduled activity comparable with the laboratory sessions in chemistry and physics courses.

(d) A viable system must have administrative data processing capabilities since there is an existing commitment here in terms of resources and personnel.

From these and other considerations, a comprehensive system has been designed. The hardware is currently available and the software will be implemented at IIT over a two year period.

The Hardware

The hardware system consists of the following:

36K 16 bit word mini-computer
900 nsec cycle time (TEMPO 1)

2 2314 type disk units
capacity 28,000,000 16 bit words

2 tape units 30 ips.

1 electrostatic printer
(700 to 5000 lpm and graphics capability)

4 synchronous communication lines

1 card reader (400 cpm)

1 DCT-500 30 cps impact printer

16 programmable CRT terminals with point graphics capability

The total cost of this system is approximately $200,000 with annual maintenance cost of approximately $25,000. The useful lifetime of the system is at least 5 years. Thus the pro-rate annual cost of the system is a reasonable percentage of the annual budget of a typical liberal arts college.

The Software

I. The Administrative Data-Processing Package

As previously stated, an important component of the Computational Laboratory is the administrative data processing package. From a hardware point of view, the proposed configuration substantially exceeds the power of a 360/30 with 65K memory. Thus, given appropriate software, its capabilities for data processing use should be adequate for many purposes, particularly in view of the fact that runs requiring a large scale processor, for example modular scheduling applications, could be processed remotely.

The ADP package consists of four components:

(a) An RPG processor for simple file maintenance applications.

(b) A small COBOL processor downward compatible from ANSI standards.

(c) A sort package for disk and tape files.

(d) An indexed sequential file management system. The logical organization provided by this type of system is ideal for many applications in which permanent files are maintained on disk.

II. The Desk Calculator

The desk calculator simulator gives the terminals the capability of being used as sophisticated electronic calculators with programming capability. This might, for example, be used to provide a highly effective substitute for the classroom of calculators sometimes used in the teaching of statistics.

An interesting feature of the calculator is that it is implemented entirely within the programmable terminals. This ensures minimum load on the processor and gives guaranteed instantaneous response. To give an idea of the level of sophistication of this package, it is possible to store the program and data to solve a 4 x 4 system of simultaneous equations by Gaussian elimination.

III. The File System

Many parts of the system are based on the concept of easily manipulated files which reside on the disk storage.

The programmable terminals are used to provide an extremely powerful editing system which is very easy to use. Basically, the screen acts as a window scanning through the file, and the information is modified on the surface of the CRT using a range of editing functions, including character insert/delete, line insert/delete, field skip/duplicate, etc.

This 'frame processing' approach to editing is powerful, easy to use, and requires very little attention from the CPU thus assuring rapid response. A similar system, NUROS, is in use at the University of Nebraska using 2260 terminals on a large IBM 360/65.

Suitable system commands allow files to be created, copied, updated, extended, saved on tape, printed, etc.

IV. Remote Access System

The Remote Access System provides comprehensive hardware and software support for accessing a variety of large scale remote computers. A processor language which is being designed allows creation of a remote entry module (REM). The REM basically performs two functions:

(a) An input file (from Disk) is converted into suitable form for use remotely (by reformatting, adding control cards, translating character sets, etc.)

(b) The received output is converted to a usable local file (by removing extraneous system messages, etc.).

V. Local Conversational Processors

A variety of 'compile and go' processors will be implemented. These will either run conversationally, with input and output directly to the terminal, or the input and output can be associated with prenamed files. In the latter case, the output could be printed on the high speed printer.

Initially, the design includes two such processors:

(a) IITRAN, the student oriented language designed and implemented at IIT.

(b) SNOBOL-4, the string processing language developed at Bell Telephone Laboratories. A recent implementation of SNOBOL-4, called SPITBOL, for the 360, done at IIT, shows this language to be practical on a machine of this size.

It is anticipated that other processors in this class will be implemented as various students become involved with the project. Based on current interest, these might include BASIC, FORTRAN, APL, LISP, ALGOL-68.

VI. The CAI Package

A language for applications in computer assisted instruction will be part of the system. Here, extensive use of the large capacity disk storage will be made for storing programs and lessons. Also, the capability of the programmable terminals will be used to provide frame oriented display including graphics.

The exact form of the CAI language has not be determined. A requirement is that it be simple for non-computer oriented faculty members to devise and implement course materials. A model of the type of language under consideration may be found in the UNIVAC COPI language which is in wide use in varied applications.

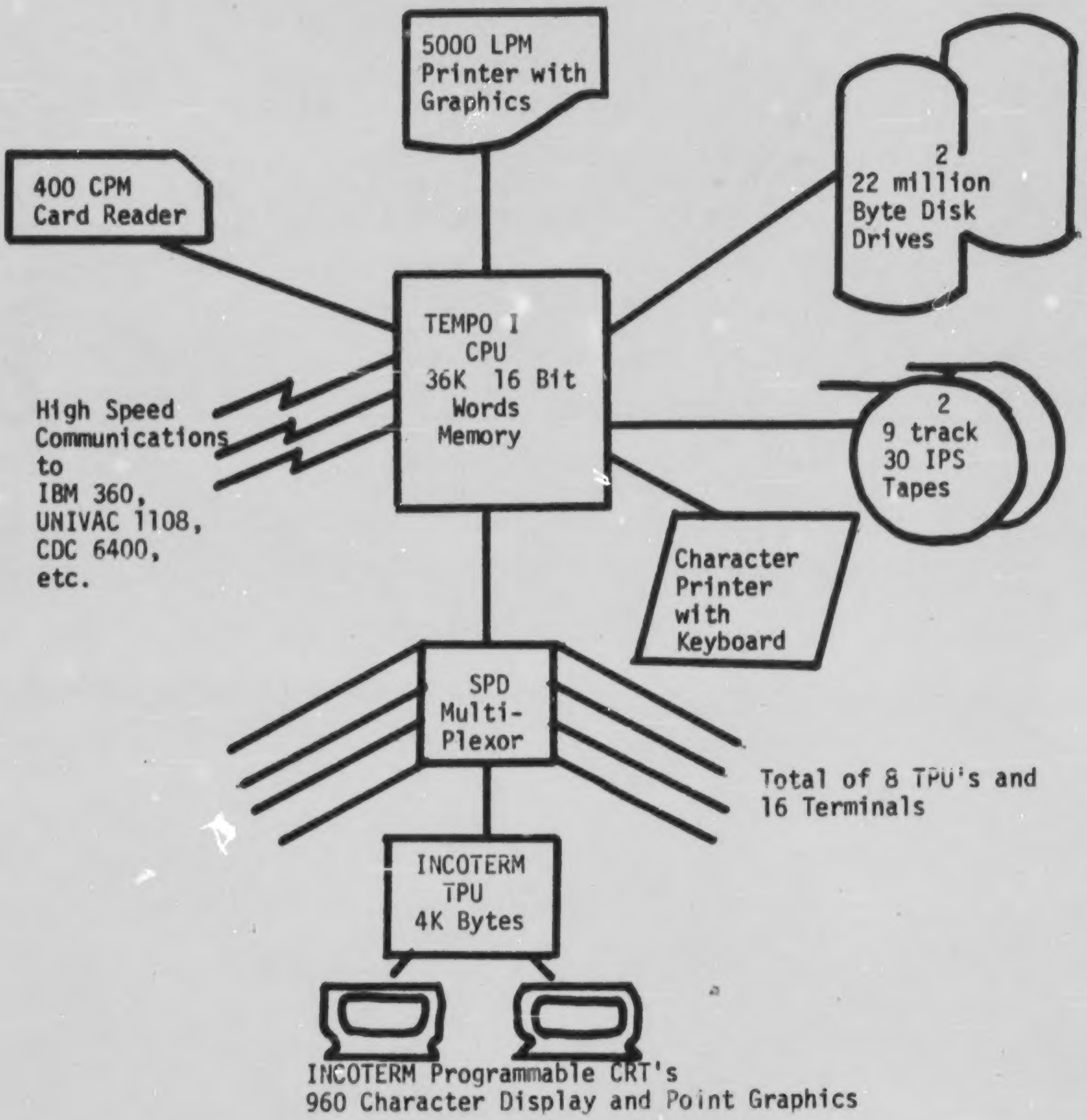
HARDWARE CONFIGUATIONS AND APPROXIMATE PRICES * (May 1, 1971)

1.	8 terminals, page printer, 1 disk, 1 tape, 24K core	$122,000
2.	12 terminals, page printer, 1 disk, 1 tape, Gould printer/plotter, 32K core	$161,000
+3.	16 terminals, page printer, 2 disks, 2 tapes, Gould printer/plotter, 36K core	$205,000
4.	24 terminals, page printer, 2 disks, 2 tapes, Gould printer/plotter, 40K core	$237,000
5.	32 terminals, page printer, 2 disks, 2 tapes, Gould printer/plotter, 48K core	$268,000
6.	48 terminals, page printer, 3 disks, 2 tapes, Gould printer/plotter, 64K core	$354,000

* These prices include hardware only, software adds about 10% (estimated)

+ The configuration illustrated on the following page and all subsequent pages refer to this configuration.

HARDWARE CONFIGURATION


5000 LPM
Printer with
Graphics
400 CPM
Card Reader
2
22 million
Byte Disk
Drives
TEMPO I
CPU
36K 16 Bit
Words
Memory
High Speed
Communications
to
IBM 360,
UNIVAC 1108,
CDC 6400,
etc.
2
9 track
30 IPS
Tapes
Character
Printer
with
Keyboard
SPD
Multi-
Plexor
Total of 8 TPU's and
16 Terminals
INCOTERM
TPU
4K Bytes
INCOTERM Programmable CRT's
960 Character Display and Point Graphics

TEMPO 1 CPU & MEMORY CHARACTERISTICS

Memory: 36K 16 bit words (72K bytes), cycle time 900nsec.

Register: 16 general purpose 16 bit xregs
10 non-programmable special purpose registers

Instructions: full set of memory to register, and register to register instructions including full set of logical and shift instructions

Addressing: 13 modes, including double indexing, indirect reference and table search

Timing: load from memory 1.8 μsec.
register-register 0.9 μsec.
mutiply/divide 6.0 μsec.

Input-Output: maximum total rate 700,000 16 bit words per second
12 nested, maskable, interrupt levels

TERMINAL SUBSYSTEM

Type: INCOTERM SPD 10/20 Dual

Display: 15 x 64 = 960 characters or 60 x 126 points (or mixture)

Two displays are controlled by one SPD 10/20 TPU. The TPU characteristics are as follows:

Program Registers: 3

Instruction Set: Includes arithmetic, logical, brancn, and I/O instructions.

Speed: Memory 1.6 μsec. Load Instruction 3.2 μsec.

Core: 2K 16 bit words (1K words used for refresh)

CARD READER SUBSYSTEM

Format: Standard 80 column cards, Hollerith or column binary

Speed: 400 c.p.m.

Capacity (hopper): 500 cards

PRINTER SUBSYSTEM

Type: Electrostatic (dot matrix)

Speed: 4,800 l.p.m. (Character lines)
~40,000 l.p.m. (dot lines)

Resolution: 80 dots per inch (full graphical capability, including special forms generation)

Character Sets: Hardware fonts for ASCII charc in 5 x 7 or 8 x 10 matrix (horizontal or vertical).
Software can print any chars using a program defined dot matrix set.

Paper Size: 11 inch (roll or fanfold)
(132 chars horizontal with 5 x 7 font)

Paper Type: Special coated paper (cost is less than 2¢ per page)

DISK SUBSYSTEM

Drives: Two - simultaneous seek capability

Format: IBM compatible fixed length sectors (346 bytes)

Access: 35 msec. average head positioning.
12.5 msec. average rotational delay

Speed: 156,000 16 bit words per second

Capacity: 11,200,000 16 bit words per drive
(total capacity 45,000,000 chars)

Packs: Standard IBM 2314 type (removable)

TAPE SUBSYSTEM

Drives: two - simultaneous rewind capability

Format: IBM compatible 9-track

Density: 800 BPI

Speed: 25 i.p.s. = 30,000 chars per second

Capacity: 2,400 ft. standard reels

COMMUNICATIONS SUBSYSTEM

Type: Synchronous

Speed: 2K baud (phone line)

A variety of channels provide sufficient flexibility to communicate with a wide range of large scale computers including IBM 360, UNIVAC 1108, CDC 6400, 6600, 7600.

7-33

The Eastern Michigan University Center for the Exchange of Chemistry Computer Programs:

EMU-CECCP

by

Ronald W. Collins
Chemistry Department
Eastern Michigan University
Ypsilanti, Michigan

Many chemistry departments are now utilizing some form of computer-aided instruction in their coursework. This use ranges from sophisticated interactive computer-assisted instruction (CAI) to the simplest form of non-interactive computer applications (NICA), namely data reduction. Regardless of the exact nature of the computer usage, however, there is an emerging need in many departments for a library of computer programs to implement their planned computer-aided instruction. Most chemistry instructors have been meeting this need by writing their own computer programs, specifically for their own pedagogical purpose and designed exclusively for their own computer system. To date communication within the academic chemistry community on such efforts has been relatively poor. This has lead to considerable duplication of effort in the writing of essentially identical programs for basically identical purposes at different institutions. Admittedly the portability of computer programs is often hampered by systems variations among different machines; however, considerable time can still be saved by eliminating repetition in developing algorithms for given chemistry applications. In an attempt to provide a service which will help to eliminate some of this needless duplication in programming, the off-line Eastern Michigan University Center for the Exchange of Chemistry Computer Programs (EMU-CECCP) has been established.

The EMU-CECCP was originated in 1970 with emphasis strictly on computer programs of an educational nature; i.e., programs which could readily be used in chemistry courses, rather than highly specialized research programs. The rationale for creating the EMU-CECCP goes beyond, however, the simple expediency of exchanging proven documented programs, to include also the hope that the center will serve to create an awareness of professional expertise in given areas of computer applications to chemical problems. In addition it is anticipated that the EMU-CECCP can, by hosting periodic conferences, serve to promote the exchange of ideas among chemists on the role of computers in chemistry education. The EMU-CECCP is at present internally financed by Eastern Michigan University, and operated via the part-time efforts of several EMU chemistry faculty and graduate students. The geographical scope of the organization has been primarily the state of Michigan along with several neighboring Midwestern states.

The EMU-CECCP does not represent a unique concept; i.e., the basic idea of a program exchange is not original to this organization. Consequently, the strength of the EMU-CECCP does not lie in its basic originality, but rather in the heavy emphasis being placed on effective indexing and documenting of all programs contained in the library. Each EMU-CECCP program is thoroughly described in a "User Description", which is an 8-10 page document covering the theory and use of that particular program. The exact format for the User Description varies from one program to another but the essential contents remain the same. Included are a flow diagram, symbol chart, complete program listing, instructions for data card preparation, and a sample problem to check the program execution.

The EMU-CECCP disseminates information regarding its inventory of programs by issuing semi-annual numerical and keyword indexes. Each program accepted for inclusion in the EMU-CECCP library is cataloged under both a program name and an assigned number, and its function is described by not more than five keywords. These "keywords" can be either English words, chemical symbols, or appropriate mathematical expressions. Both an alphabetical keyword index and a numerical list of all available programs is then published. To date these indexes have been supplied gratis to all 63 institutions of higher learning in the state of Michigan, plus other interested individuals who have been referred to our organization. The suggested sequence for efficient use of these index materials is as follows:

1. Consult the alphabetical keyword index (Figure 1) for terms indicative of the application you have in mind; note the program number(s) corresponding to these terms.
2. Consult the numerical list of programs (Figure 2) under the appropriate number(s) to obtain more information about those programs which appear to be of interest to you. This list describes the input and output, as well as the programming language for each computer program.
3. Based on this additional information, if a given program(s) is of definite interest to you, request from the EMU-CECCP the user description for that program(s). A form for such requests is supplied for the convenience of all EMU-CECCP subscribers.

At present the function of the EMU-CECCP terminates at this point; i.e., the organization is not supplying duplicate source decks for programs (requests can be directed to the program author, but not to the EMU-CECCP) nor is it attempting to become involved in problems of systems capability. The EMU-CECCP does, however, serve as many users as possible by including in our library essentially duplicate programs written in different programming languages.

The organizational format of the EMU-CECCP seems to be both efficient and reasonably effective in meeting the needs of our various users; however, several problems have arisen. First, program authors must write the user descriptions for their own programs rather than depending on the EMU-CECCP staff to prepare these documents. Originally this was not the plan, but it has been shown rather conclusively that an author can document his own program in considerably less time, thereby facilitating more rapid accessibility to his programs via their earlier inclusion in the EMU-CECCP index. A second problem has been communicating with potential program donors who might wish to make their computer programs available to other chemists but who are not aware of the EMU-CECCP. For this reason the EMU-CECCP is being extensively publicized via national and regional meetings of the American Chemical Society as well as through specialized computer-oriented conferences.

Regarding other future plans, the EMU-CECCP will continue to be only an off-line source of computer programs for chemistry education; i.e., there are no plans to provide user technology in the form of on-line access to our program library via a time-shared network operated from the EMU campus. It is our contention that efficient, economical time-sharing is available through a variety of sources, and that the efforts of the EMU-CECCP staff are better directed toward developing and operating the best possible off-line educational program exchange. A review of our current library holdings does reveal, however, that the majority of the EMU-CECCP programs are written for batch processing. Thus, an attempt is currently being made to solicit both conversational interactive CAI programs, and pseudo-interactive programs whose input and output formats are compatible with remote teletype terminal operation.

EMU - CECCP

Keyword Index to Computer Programs

(listed by program number; entries in capital letters are program names rather than keywords)

ACETIC ACID - 6
Acid, weak - 25
Ammonia, aqueous - 24
ANISTROPY - 13

Base, strong - 25
Benzene, derivatives - 14, 15

Coefficients, distribution - 30
Coefficients, equilibrium - 30
Coefficients, extinction - 16
Complexes, spectra - 32
Constant, equilibrium - 7
Constant, ionization - 6, 8
Constant, stability - 16
Constant, Van der Waals - 21
Correction, calorimetric - 11
Curve fitting - 20

DELH - 17
Derivative, method - 1, 28
DERTI - 1
Deviation, percent - 29
DIPOLE FIELD - 12
Diprotic acid - 8
DJPR - 30
DSPAC - 18
DSPACE - DELSQ - 3
D-Spacings - 3, 4, 5, 18

EDTA: OBJ - 26
Electrode potential - 24
Electronic absorption - 32
Energy, transition - 32
ENTHALPY - 11
EQUICON-FESCN - 7
Equilibrium constant - 7
Extinction coefficient - 16

Field shift - 12

Guggenheim method - 23

HA: OBJ - 25
Heat capacity - 11
H_2O_2, decomposition of - 23
$[H^+]$ - 10

INTERPLANAR - 4
IONICON - 8

Kinetics - 2, 31
KSPAGAC - 9

Least squares, curve fitting - 19, 20, 27
Least squares, errors - 19
Least squares, linear - 20
Least squares - 19, 20, 27
Least squares, weighted - 17
Ligand field - 32
LINLEAST - 19
LSQ: FOR - 27

McWeeny method - 14
Miller indices, random - 5
MILLER SPACINGS - 5
Molal volume - 22
Monoprotic acid - 8
M^{+2} - 26

Nernst, eq. - 24
NERNST: OBJ - 24

OHCO - 32

Polynomials - 27
Position number - 12

RATE - 23
Rate constants - 2, 23, 31
REGRESSION - 20
RING CURRENT (I) - 14
RING CURRENT (II) - 15
Rotating bond - 12
Rotational increment - 13

FIGURE 1

EMU - CECCP

Library of Computer Programs

Number	Program Name	Author and Institution	Key Words	Input	Output	Language and Machine	Date of Inclusion
1	DERTI	R. W. Collins EMU	Titration Deriv. Method	Volumes, pH or Volts	End point conc, %	FORTRAN IV IBM 1130	10-1-70
2	SVMY	Sudhir Vaidya EMU	Kinetics Rate Consts.	Time in Sec.	Average Rate Consts.	FORTRAN IV IBM 1130	10-1-70
3	DSPACE-DELSQ	R. W. Collins EMU	X-Ray Diff. D-spacings $\Delta \sin^2\theta$	Arc-Measurements Intensities	D-spacings $\sin\theta$ $\Delta \sin^2\theta$	FORTRAN IV IBM 1130	10-1-70
4	INTER-PLANAR	Sudhir Vaidya EMU	D-spacings Unit Cell Volume	a, b, c α, β, γ h, k, ℓ	D-spacings Unit Cell Volume	FORTRAN IV IBM 1130	10-1-70
5	MILLER-SPACINGS	Sudhir Vaidya EMU	D-spacings Random Miller Indices	a, b, c α, β, γ	Random h, k, ℓ D-spacings Unit Cell Volume	FORTRAN IV IBM 1130	10-1-70
6	ACETIC ACID	Namie Smith EMU	Ionization Constant	Normality B FW, V, pH	K_{AVG}	FORTRAN IV IBM 1130	10-1-70
7	EQUICON FESCN	Sherman MacCready Jackson Community College (Mich.)	Equilibrium Constant	$[Fe^{+++}]$,$[SCN^-]$ % Transmission	Equilibrium Constant	FORTRAN IV GE 430	10-1-70
8	IONICON	Sherman MacCready Jackson Community College (Mich.)	Ionization Constant Monoprotic Acid Diprotic Acid Weak Acid	Vol. Acid Vol. at Avg. End Point pH	Avg. Ionization Constant	FORTRAN IV GE 430	10-1-70

FIGURE 2

THE USE OF APL IN THE UNDERGRADUATE CHEMISTRY CURRICULUM

Thomas R. Dehner

Bruce E. Norcross

State University of New York at Binghamton

Abstract

For the past few years members of various departments at the State University of New York at Binghamton have been exploring the applicability of APL (A Programming Language) to instructional use in the undergraduate curriculum. APL is a particularly attractive language for computer-assisted instruction because: 1) it permits direct and immediate interaction between the student and the computer; 2) the student can learn to use the language with a minimum of instruction; and 3) it not only allows programming in the conversational mode, but also has considerable computing, logical, and plotting capabilities. This paper outlines our experience with APL in the chemistry curriculum. In particular, a fairly involved program used by our students in the General Chemistry Laboratory experiment, Molar Volume of a Gas, is described in some detail. Included is a discussion of our objectives and of student reaction.

Introduction:

The past few years have witnessed a greatly increased effort to realize some of the educational potential presented by the development of computers. A number of different systems and languages have been tried in computer-aided instruction, with varying degrees of success and promise. One of the most promising languages, in our view, is APL (1), a programming language which allows direct interaction between the user and the computer by typewriter communication.

Recently a grant from the IBM Company permitted an exploration of the applicability of APL to instructional use on the college level at our institution. Since then, faculty members in chemistry, physics, biology, geology, mathematics, philosophy, and music, have taken advantage of this opportunity. Although frustrated at times with the amount of time required to develop, write, and de-bug useful programs, those involved in this exploration have for the most part remained enthusiastic about the use of computers in undergraduate teaching, and about APL as a useful language for computer-assisted instruction. In this paper some of our efforts and experiences in the use of APL in chemistry courses at the State University of New York at Binghamton will be described.

It is easy to be enthusiastic initially about using computers in teaching. The computer may indeed provide a new dimension in teaching, although care should be used in its application (2). Furthermore familiarity with that "mechanical monster" which may unwittingly send bill after bill, month after month, despite desparate pleas and claims of payment, and which has the awesome power of ruining otherwise perfect credit ratings, is certainly desirable for both teacher and student! It is also probably advantageous for students in science to become acquainted with the computer and its applications at an early stage in their training.

A variety of program types for instructional use has been suggested (3). These include:

(a) programs for data reduction and statistical treatment of experimental data;

(b) the simulated lab program, which allows the student to concentrate on the interdependence of physical variables or to examine parameter relationships in physical models;

(c) remedial drill programs or extensive tutorial programs in which the computer is the impersonal and infinitely patient instructor, allowing the student to progress without embarrassment and at his own pace through particular subject material, all the while wisely questioning the student to test his understanding of the material. This last type is undoubtedly the most difficult kind of program to write.

But, as most of you know, it's a long road from such visions to useable programs!

APL Language:

Even before defining firm objectives for efforts in CAI one must consider the computing system available and the choice of program language. Ideally, the language should be: 1) one which is easy to program and easy to use, 2) one which allows direct and immediate communication between the user and the computer, 3) one which permits extended conversational as well as computational operations, and 4) one which accommodates auxilliary interactive facilities such as slides, tapes, and CRT display. Perhaps APL, more than any other current language, comes closest to this ideal.

APL, A Programming Language (1), was first developed and used at IBM in the early 1960's. It is an interactive, algorithm-oriented time-sharing language with a single, consistent notation for many areas of discourse. By means of the typewriter terminal the user can define and store programs and execute system commands or mathematical statements, and receive an almost immediate response from the computer. The primitive functions and operations in APL offer great flexibility and simplicity in mathematical and logical operations - particularly in manipulation of vectors and matrices. Not only can the language handle fairly sophisticated calculations and mathematical operations, it is also relatively easy to program. In addition, APL has a graphic display capability in its plot functions, and allows programming in the extended conversational mode so that instructions may be given, and questions asked, in alphabetical statements as well as in mathematical formulations. Finally, programs can be easily edited and modified, and can be stored in public libraries for ready access by any user. No class time is needed for computer instruction. Brief handout instructions, supplemented by individual instruction at the terminal, have been sufficient to allow the student to use the programs which have been made available. At State University of New York at Binghamton, those students who are interested can attend a set of video-taped lectures on APL which are shown several times a year. Because of the relative simplicity of the language, a student (or faculty member) who has attended only a few of these lectures can begin to write his own programs.

We have experimented with the use of APL in the Chemistry Curriculum at every undergraduate and graduate level. Program types include laboratory simulation, laboratory data reduction, and statistical treatment of experimental data (4). We have yet to complete a tutorial-type program in chemistry, although we are currently working on one on the mole concept and stoichiometry.

Initially we were operating with an IBM 360/40 computer with a limited number of terminals. APL was available only three hours per day. Consequently, only a fairly small number of students were involved in the use of our programs. (It is interesting to note that the terminals were in use 90% of the time when APL was available, compared to approximately 10% use during remote FORTRAN hours.) Recently we have acquired an IBM 360/67 computer, and APL is now available on campus eight hours per day. Currently we have 24 remote typewriter terminals (#2741) on campus and seven off-campus. This year, for the first time, all of the students in general chemistry (over 300) used the computer.

General Chemistry Programs:

Lab Simulation: Initially students in one of the second semester freshman chemistry recitation sections worked with two titration programs developed by Science Research Associates (Chicago, Illinois), and somewhat modified by us. These programs allow the students to simulate acid-base titrations for a variety of weak acids, and turbidimetric titrations to obtain solubility product information for a number of slightly soluble salts.

The program TITRATE randomly chooses a pK within a specified range and asks the student to specify his initial titration parameters - i.e., molar concentration of both acid and base and volume of acid. The student then adds aliquots of base, and the computer calculates the pH after each addition. If an error is made during the course of the titration (ex.: too few points in the region of the endpoint) a reset to zero conditions is easily made, and the titration is repeated. The titration record is saved, and at the end the student can choose to see a graph of pH vs. volume of titrant for his titration. We have recently added options for first and second derivative graphs, expanded in the region of the endpoint. We plan to expand this program to include treatment of polyprotic acids and buffer solutions.

The program KSP extends the student's experience with equilibrium systems, again through simulated titration experiment. In this case the endpoint is indicated by "lasting turbidity upon shaking", and the KSP is calculated from the titration data.

Response from the limited number of students involved with these programs was enthusiastic. (In fact several volunteers were coming at 7:00 A.M. in order to use the computer!) Although this type of program does not provide experience with lab technique, it can help the student grasp the basic concepts involved in titrimetric analysis.

Statistical Treatment of Data:

Later in the year a set of simple kinetics programs was made available to introduce the students to statistical treatment of experimental data. The student supplies the data (either from assigned problems or from laboratory experiment) together with an estimate of error in both concentration and time values. The kinetics programs apply simple first order and second order treatment to the data, calculate the best straight line fit for the data by a simple least squares treatment, and print out the rate constant calculated from the slope, and the standard deviation. Again, the student can choose to see both first and second order graphs of the data and can compare the rate constant value from the least squares treatment to the one he has obtained from his own hand-drawn graph of the same data.

An additional program in this set is used for the calculation of the activation energy from the temperature dependence of the rate constant. In this program the student can investigate the effect of error limits on the activation energy and standard deviation. Initially the computer calculates the best fit by a simple least squares method and prints out the activation energy from the slope of this line as its first approximation. The results of this calculation are then used to solve cubic equations from a more sophisticated line fitting treatment by a series of successive approximations. When successive values differ by less than 1% the calculation is terminated and the standard deviation is printed.

Again, only a small number of students used these programs, and we have as yet no valid measurement of their utility. At the very least we can say that these students were introduced early to one of the important uses of the computer for experimental data treatment, and to the fact that experimental results should always include some indication of estimated or calculated experimental error.

Laboratory Data Evaluation and Reduction:

Our most ambitious program at the freshman level is one which analyzes and stores experimental data obtained by the student in the laboratory determination of the molar volume of oxygen. The purpose of this laboratory experiment is to teach the student quantitative techniques in working with gases and to verify Avogadro's hypothesis concerning gas behavior. The student weighs out a sample of a mixture of potassium chloride (KCl) and potassium chlorate ($KClO_3$) (% Unknown) and, with appropriate apparatus which he has assembled, decomposes the $KClO_3$ by heating, and collects and measures the oxygen gas evolved. From the primary data collected (weights, volume of water displaced, temperature, barometer reading) the student can then calculate a molar volume for oxygen at standard temperature and pressure and the % $KClO_3$ in his sample.

The experimental procedure is a fairly demanding one for students at this level, and there is plenty of opportunity for experimental error. Six pieces of primary data must be obtained. On these the student performs a number of calculations, including aqueous vapor pressure and barometer scale expansion corrections, before he arrives at his final answer. Typically, a student would carry out one determination and just finish his calculations by the end of the lab period, only to find that his final answer is unacceptable and he must repeat the experiment. Clearly, it would be helpful to the student to have an immediate check on the accuracy of his primary data.

The objectives of the computer programs in the CHEMLAB workspace are:

1) To allow the student to check immediately the accuracy of his experimental measurement of the volume of oxygen;
2) To allow the student to check his calculations of molar volume and % composition for his particular sample;
3) To store student primary data and results for subsequent analysis and grading.

The initial impetus for this program come from a paper presented at the Symposium on the Status of Computer-Assisted Instruction in Chemistry, held at Toronto in May, 1970 (5). In this paper, D.S. Olson and colleagues described the use of computer punch cards for grading chemistry laboratory reports for large classes at the Air Force Academy. We realized that by using an interactive language such as APL, and by placing computer terminals in the laboratory, we could not only ease the grading burden for our teaching assistants and collect data for class analysis, but, more importantly, could also permit the student to distinguish between experimental error and calculation error in data treatment while his apparatus was still available for repeats.

As soon as the student has finished collecting data for his first run, he goes to the computer terminal in the lab and types CHEMLAB1, which starts function execution. (Just prior to the laboratory period the lab instructor has signed onto the computer and loaded a copy of the CHEMLAB workspace from the public library into his active workspace by a simple system command. Three additional system commands initialize the program so it is ready for use by students in his lab section.) This function has three main parts. Access to each part is determined by a sequencing code in the program. The first part provides for entry of primary experimental data. The student is instructed to enter his name, sample designation, weights, temperature, barometer reading, and collected volume of oxygen. The program then calculates the theoretical volume of oxygen (using student's conditions of temperature and pressure, and using % composition information stored in program for the designated sample) and compares it with the student's experimental value.* The computer response is determined by the agreement between these

* (See appendix for sample execution of CHEMLAB.)

values. If the student's data is good, he is told to proceed with his calculations. His name and data are stored in the workspace and the sequencing code is advanced, so that the next time the same student calls for the program, it automatically branches to Part 2. If the agreement is fairly good, the student is allowed to proceed with his calculations, but is advised to repeat the experiment. If the agreement is poor, then the student is told that his data is unacceptable and that he must repeat the experiment. In the case of fair or poor results, leading questions are asked by the computer to help the student pinpoint likely sources of error.

Once his data is accepted by the computer, the student can then, from this primary data, calculate the weight of oxygen, the partial pressure of oxygen, and the molar volume of oxygen at STP. Part 2 of CHEMLAB1 requires the student to enter his calculated values. The program then does the same calculations, using the student's primary data which is stored in the DATA matrix and the correction factors stored in the program. The computer also calculates the theoretical values for the particular sample which the student has analyzed. The three columns of data are then printed out side by side for comparison.* This allows the student to check immediately a) the validity of his calculations, and b) the accuracy of his experimental measurements. The student includes this part of the computer printout in his lab report, and is expected to discuss the agreement or disagreement of the three columns.

Again, if the student's data is acceptable he may proceed to Part 3. From the experimental weight of oxygen, the known molecular weight of O_2, the total sample weight, and the balanced equation for the reaction, he can calculate the percent potassium chlorate in his sample. He then enters the results of this calculation in Part 3. The program does the same calculations and prints out the two columns of data, along with the theoretical % $KClO_3$ for the student's sample. Again, the student can check both the validity of his calculations and the accuracy of his results.

---

* See appendix.

Once the student's data is accepted and saved by the computer, it is stored in the workspace and cannot be altered. The program LABINSTRUCT permits the laboratory instructor to retrieve all of the names and data from his lab section in a defined format for comparison, analysis, and grading.

This program was initially written in the summer of 1970 and first used by a small class in that summer session of general chemistry. The experience with the summer group indicated that the program was useful and led to fairly extensive modifications and expansion of the program. The revised program was used by over 300 students in the fall semester of 1970. Although a few students regarded the use of the computer as an extra chore, the general response of the students can be described only as enthusiastic and excited. Students suggested further modifications of the program to make it more helpful. This led to further revisions which were incorporated for use in the spring semester. Students also suggested that similar programs for other experiments would be very helpful.

The program also received strong support from the lab instructors. Usually, for this experiment, if the final calculated results are in error, the instructor must refigure each calculation until the error is discovered. It is therefore quite a chore to evaluate constructively and conscientiously 40-50 lab reports. The computer program eliminates all of the "busy work" for the instructor and allows him to pinpoint the source of error very readily.

In many cases, because the student can also check his own errors, he can discuss his experimental results more intelligently.

Finally, ready access to primary data for the entire large class permitted us to examine closely the results of the experiment and to set reasonable limits of acceptability.

The CHEMLAB program has been described in some detail because it is our most successful program to date. We are encouraged by its reception and are convinced that this is a useful way to use the computer in the laboratory. The program continues to develop with each class. Future intended modifications include tutorial-type questions to guide the student into more thoughtful analysis of the experiment.

Advanced Courses:

APL programs have also been used in some of the advanced courses in the chemistry curriculum. For example, in the organic laboratory a program is used to calculate the percent composition of liquid mixtures from vapor phase chromatography data. In the introductory physical chemistry course a program has been written (by a student) which calculates the eigenvalues and wavefunctions for a one-sided potential well. By varying the parameters (depth and width of well and mass of particle) the student can follow the effect of parameter variations on the eigenvalues and attempt to discover valid relationships.

The program for determination of activation energy, mentioned earlier, was actually developed for use by juniors and seniors in a physical chemistry lab experiment on the internal rotation of amides, measured by a nuclear magnetic resonance method. The data obtained are not susceptible to simple least squares treatment, so the students use a computer program to calculate weighting functions for statistical treatment of their NMR data (6).

Finally, in the physical organic courses, students have been required to develop their own line-fitting program in APL for treatment of kinetics and other experimental data.

Other Departments:

Several other departments at State University of New York at Binghamton have also been experimenting with APL in their courses. In the genetics and biostatistics courses fairly simple programs have proven to be both fascinating and useful to the student (7). The program RANDOMDRIFT is an example of a computer-simulated experiment in genetics which could not possibly be done by the students in the laboratory. If a population is limited in size, or if only a small fraction of the population is responsible for the offspring of the next generation, the genetic composition of the population will fluctuate in a random manner. The smaller the population, the greater the random fluctuation. This is known as the Sewall-Wright Effect. Instead of merely being informed of this experimental observation, the student is permitted to experience the sensation of discovery by using an APL function which is programmed so that half of the population in each generation is randomly selected to be parents for the next generation. The student starts with four allelic genes, selects the size of the population sample and continues the reproductive process until three of the four loci are lost from the population by random drift.

Extended tutorial programs which simulate physical laws otherwise difficult for the student to investigate have been most successful in the physics department (8). Prior to the simulated experiment a series of questions in multiple choice format familiarizes the student with the problem and provides a guide to intelligent selection of parameters. Wrong choices lead to appropriate branches in the program which contain additional drill and guidance. The student specifies all important parameters for the experiment, and observes the results, usually in graphical form. Additional tutorial questions guide the student's analysis of the results and help him discover the parameter relationships. For example, the program ORBIT plots the orbit of a satellite launched tangentially from a central body. The student specifies the central body and its mass, the satellite mass, and initial distance and velocity. He then observes the effects of varying these parameters on the orbit of the satellite. Other tutorials in physics include an investigation of wave behavior and interference patterns, and analysis of Michelson-Morley type experiments to arrive at postulates of relativity. Indeed, these programs have been so successful that the physics department this year has replaced the introductory physics laboratory with a "structured combination of computer laboratory and hands-on demonstrations". The traditional introductory lab has been upgraded and moved to the sophomore year.

Conclusion:

Although our experience so far is insufficient to make extensive judgements about the educational value of our efforts in computer-assisted instruction, there are a number of very encouraging aspects of our exploration to date:

1) We have developed some programs which appear to be useful and unique as instructional aids and which can be successfully integrated into our course programs. The more workable model programs

we have on hand, the easier it is to formulate and develop additional programs.

2) We are succeeding in introducing a significant number of students to the computer and its potential uses, and they are responding enthusiastically.

3) A significant number of students, on their own initiative, are writing their own programs in APL and continuing to use the computer for course work, lab data treatment, and research data analysis.

These are the encouraging factors. The discouraging ones are the cost in both time and money to support and develop such a program. These cost factors are difficult to determine, since these programs undergo almost continual evolution even after considerable use. Recent conservative estimates at our institution (8) indicate that the development of a long tutorial like ORBIT, or an involved program like CHEMLAB, requires approximately 100-120 hours of full time effort. Thus the cost for author's time would be in the range of $1000 for such a program. The only other cost to the department in the development of the program is CPU time, at $100/hour. In most cases this is small compared to the cost of the author's time.

The major costs in a CAI program are terminal rental, at roughly $1500 per year per terminal, and student-generated CPU time. A typical calculation (assuming five terminals, an average student connect time of an hour a week, and an average 20:1 ratio of connect time to CPU time) yields an estimated cost of $20,000 per year for a class of 100 students, or roughly $200 per student. Obviously, if the cost remains at this level, one must decide whether it is more useful to have the computer available, or to hire another faculty member (or two)!

At present however, given the necessary funds, a willing and able computer center staff and the continued interest of both faculty and students, we plan to continue to work on the development of programs for computer-assisted instruction.

Acknowledgements:

The authors wish to thank the staff of the computer center at State University of New York at Binghamton for their cooperation and assistance. We are particularly indebted to Mr. James Higgins, academic manager of the computer center; to Mr. Kevin Kelly, who did most of the initial programming for CHEMLAB, and to Mrs. Anne Kellerman who has helped with the revision of CHEMLAB. We also gratefully acknowledge the financial support of the IBM Corporation.

References

1. K. Iverson, A Programming Language, John Wiley and Sons, Inc., New York, 1962; K. Iverson and A. Falkoff, APL/360 Users Manual, International Business Machine, New York, 1968; P. Berry, APL/360 Primer, International Business Machine, New York, 1970.
2. J.A. Young, J. Chem. Educ., 47:11, 758 (1970).
3. (a) "Computers in Undergraduate Education: Mathematics, Physics, Statistics and Chemistry", proceedings of Conference at University of Maryland, December 8-9, 1967.
    (b) R.C. Atkinson and H.A. Wilson, "Computers in Chemical Education", Science, 162, 73 (1968).
    (c) "The March on Armonk", proceedings of the APL Users Conference, State University of New York at Binghamton, July 11-12, 1969.
    (d) F.D. Tabbutt, "Computers in Chemical Education", Chem. & Eng. News, January 19, 1970, p. 44.
    (e) Computer Instruction Issue, J. Chem. Educ., 47:2 (1970).
    (f) Symposium on the Status of Computer-Assisted Instruction in Chemistry, Joint Conference of the Chemical Institute of Canada with the American Chemical Society, Toronto, Canada, May 27, 1970.
4. T.R. Dehner and B.E. Norcross,"The Use of APL (A Programming Language) in Computer-Assisted Instruction in Freshman Chemistry", presented at 158th National American Chemical Society Meeting, New York, September 8, 1969.

5. D.S. Olson, S.R. Carroll, L.A. King, J.F. Altenburg, Symposium on the Status of Computer-Assisted Instruction in Chemistry, Joint Conference of the Chemical Institute of Canada with the American Chemical Society, Toronto, Canda, May 27, 1970.

6. B.E. Norcross, R.H. Cartmell, G. Hall, "The Hindered Rotation of Amides", presented at 158th National American Chemical Society Meeting, New York, September 8, 1969.

7. A.P. Mueller, J. Biol. Ed., 3:2, 159 (1969); Proceedings of the APL Users Conference, State University of New York at Binghamton, July 11-12, 1969.

8. C. Stannard, Am. J. of Physics, 38:12, 1416 (1970).

Appendix

Figure Number	Title
1	Beginning portion of APL program, CHEMLAB1.
2	Portion of APL program, CHEMLAB1.
3	Sample execution of CHEMLAB1, part 1.
4	Sample execution of CHEMLAB1, part 2.
5	Sample execution of CHEMLAB1, part 3.
6	Sample printout of stored data from CHEMLAB1, by program INSTRUCTOR.

```
      ∇CHEMLAB1[⎕]∇
    ∇ CHEMLAB1
[1]    ''
[2]    ''
[3]   START:'       TYPE IN YOUR NAME - FIRST NAME FOLLOWED BY LAST NAME.  BE SURE TO TYPE IN YOUR NAME'
[4]    '  EXACTLY THE SAME WAY EACH TIME YOU USE THIS PROGRAM.'
[5]    ''
[6]    YES←1+NO←I←0
[7]    INPUT←((⌊/(INPUT≠' ')/⍳15)-1)⌽INPUT←15↑⍞,15⍴' '
[8]   ON:→(((I<1↑⍴DATA)∧((~IO)×~ID)),(IO←×/INPUT=NAME[I;]),(ID←×/NAME[I←I+1;]∊' '))/ON,MATCH,FILL
[9]    'NAME TABLE FILLED.'
[10]   →0
[11]  FILL:NAME[I;]←INPUT
[12]   ''
[13]   '       WHEN YOU ENTER DATA, ALL WEIGHTS SHOULD BE IN GRAMS, ALL VOLUMES IN MILLILITERS, AND'
[14]   ' ALL PRESSURES IN MILLIMETERS[OF HG], UNLESS SPECIFIED OTHERWISE.
[15]   ''
[16]   '       DO NOT INCLUDE UNITS WHEN ENTERING DATA.'
[17]   ''
[18]   '      DO YOU NEED ADDITIONAL INSTRUCTIONS?'
[19]   →(~⎕)/MATCH
[20]   ''
[21]   '       DATA IS ENTERED BY TYPING IN THE DESIRED CHARACTERS AND PRESSING THE ''RETURN'' KEY.'
[22]   '  IF A MISTAKE IS MADE IN TYPING, DO NOT PRESS THE ''RETURN'' KEY; USE THE ''BACKSPACE'' KEY TO'
[23]   '  BACKSPACE THE PRINTING ELEMENT UNTIL IT IS POSITIONED DIRECTLY OVER THE FIRST (ON LEFT) FAULTY'
[24]   '  CHARACTER.  DEPRESS THE ''ATTN'' KEY AND REPLACE THE FAULTY CHARACTER(S) BY TYPING IN THE CORRECT'
[25]   '  CHARACTER(S).  RETYPE ALL CHARACTERS TO THE RIGHT OF THE CHARACTER(S) REPLACED.  PRESS THE ''RETURN'' KEY.'
[26]   ''
[27]   '       IF A MISTAKE HAS ALREADY BEEN ENTERED BY PRESSING ''RETURN'', CONTINUE ON UNTIL I ASK IF ANY'
[28]   '  MISTAKES HAVE BEEN MADE.'
[29]   ''
[30]  MATCH:''
[31]   '       ANSWER THE NEXT TWO QUESTIONS BY TYPING EITHER 1 OR 2 .'
[32]   ''
[33]   '      IS THIS TRIAL 1 OR TRIAL 2 ?'
[34]   SWITCH←⎕
[35]   '      IS THIS SAMPLE:  [1] PURE KCLO3  OR   [2]  AN UNKNOWN COMPOSITION OF KCLO3 ?'
[36]   NUMBAN←1
[37]   →(⎕=2)/GOON
[38]   '      ENTER THE UNKNOWN SAMPLE DESIGNATION.'
[39]   NUMBAN←NUMBAN+'ABCDEFGHIJKL'⍳1↑(((A≠' ')/⍳10)-1)⌽A←10↑⍞,9⍴' '
[40]  GOON:DATA[I;SWITCH;1]←NUMBAN
[41]   →((⍳4)∊DATA[I;SWITCH;15])/FAR,FURTHER,FINALE,FAREST
[42]  FAREST:'       MY RECORDS INDICATE YOU HAVE ALREADY ENTERED VALID INFORMATION FOR THE SAMPLE'
```

```
[58]  '|3|   WEIGHT OF EMPTY TUBE.'
[59]  DATA[I;SWITCH;4]←⎕
[60]  →(⌶26)+FUDGY
[61]  RETURN←RETURN,(⌶26)+1
[62]  '|4|   TEMPERATURE OF THE OXYGEN [CENTIGRADE]:'
[63]  DATA[I;SWITCH;5]←⌊TEMP←⎕
[64]  →(0.5>TEMP-DATA[I;SWITCH;5])/AROUND
[65]  DATA[I;SWITCH;5]←DATA[I;SWITCH;5]+1
[66] AROUND:→(⌶26)+FUDGY
[67]  RETURN←RETURN,(⌶26)+1
[68]  '|5|   BAROMETRIC READING; UNCORRECTED AT ';DATA[I;SWITCH;5];'∘ CENTI GRADE.'
[69]  DATA[I;SWITCH;8]←⎕
[70]  →(⌶26)+FUDGY
[71]  RETURN←RETURN,(⌶26)+1
[72]  '|6|   VOLUME (UNCORRECTED) OF OXYGEN COLLECTED.'
[73]  DATA[I;SWITCH;7]←⎕
[74]  →(⌶26)+FUDGY
[75]  RETURN←RETURN,L̲A̲S̲T̲←(⌶26)+1
[76]
[77]  '      HAVE YOU MADE ANY ERRORS IN ENTERING YOUR DATA ?'
[78]  →(~⎕)/(P̲T̲R̲=⍳3)/GOTCHA,GOAL,GOLLY
[79]  ''
[80]  '      ENTER THE NUMBER OR NUMBERS ( FOUND IN THE | | ) OF THE INPUT STATEMENTS ASSOCIATED WITH'
[81]  '  THE DATA YOU WISH TO CORRECT.  IF MORE THAN ONE NUMBER IS TO BE ENTERED, SEPARATE WITH BLANKS.'
[82]  GOTO←0,(((⍳(D̲Q̲←ρRETURN)-1)∊⎕),1)/RETURN
[83]  GATE←(2,D̲Q̲)ρRETURN,1⌽(RETURN-2)
[84]  →((ρGOTO)≤2)/RETURN[D̲Q̲]
[85]  GOTO←1↓GOTO
[86]  FUDGY←(⌶26)-GATE[2;(GATE[1;]∊GOTO[1])/⍳D̲Q̲]+2
[87]  →GOTO[1]
[88] GOTCHA:PART←((20×⌊0.5+DATA[I;SWITCH;8]÷20)-620)÷20
[89]  PART←DATA[I;SWITCH;8]-TCTABLE[DATA[I;SWITCH;5]-15;PART]+0.03
[90]  PART←PART-VPTABLE[DATA[I;SWITCH;5]-15]
[91]  DATA[I;SWITCH;6]←(PERCENT[NUMBAH]×(DATA[I;SWITCH;2]-DATA[I;SWITCH;4])×25528020×(DATA[I;SWITCH;5]+
      273.16))
[92]  DATA[I;SWITCH;6]←DATA[I;SWITCH;6]÷33475.758×PART
[93]  FUDGE←|(DATA[I;SWITCH;6]-DATA[I;SWITCH;7])÷DATA[I;SWITCH;6]
[94]  ''
[95]  →(FUDGE<0.1,((0.1≤FUDGE)∧(FUDGE≤0.2)),((0.2≤FUDGE)∧(FUDGE≤0.3)),(FUDGE>0.3))/VG,RG,PG,UA
[96] VG:'       YOUR EXPERIMENTAL VOLUME IS IN VERY GOOD AGREEMENT WITH THE THEORETICAL VOLUME(UNCORRECTED)'
[97]  '  FOR YOUR SAMPLE AND YOUR EXPERIMENTAL CONDITIONS. PROCEED WITH YOUR CALCULATIONS.'
[98]  →SNUDGE
```

```
CHEMLAB1

    TYPE IN YOUR NAME - FIRST NAME FOLLOWED BY LAST NAME.  BE SURE TO TYPE IN YOUR NAME
EXACTLY THE SAME WAY EACH TIME YOU USE THIS PROGRAM.

BEBORC

         ANSWER THE NEXT TWO QUESTIONS BY TYPING EITHER 1 OR 2 .

        IS THIS TRIAL 1 OR TRIAL 2 ?
[]:
        2
        IS THIS SAMPLE:  [1] PURE KCLO3   OR    [2]  AN UNKNOWN COMPOSITION OF KCLO3 ?
[]:
        2
        ENTER THE UNKNOWN SAMPLE DESIGNATION.
A
  PART 1:  ENTER THE FOLLOWING PIECES OF DATA:

|1|     WEIGHT OF TUBE AND CONTENTS BEFORE HEATING.
[]:
        35.1220
|2|     WEIGHT OF TUBE AND RESIDUE AFTER HEATING.
[]:
        34.7420
|3|     WEIGHT OF EMPTY TUBE.
[]:
        32.1770
|4|     TEMPERATURE OF THE OXYGEN [CENTIGRADE]:
[]:
        27
|5|     BAROMETRIC READING; UNCORRECTED AT 27° CENTIGRADE.
[]:
        744.3
|6|     VOLUME (UNCORRECTED) OF OXYGEN COLLECTED.
[]:
        350

        HAVE YOU MADE ANY ERRORS IN ENTERING YOUR DATA ?
[]:
        NO

    YOUR EXPERIMENTAL VOLUME, ALTHOUGH ACCEPTABLE IS IN FAIRLY POOR AGREEMENT WITH THE
THEORETICAL VOLUME(UNCORRECTED)FOR YOUR SAMPLE AND YOUR CONDITIONS. YOU MAY
PROCEED WITH YOUR CALCULATIONS ON THIS SAMPLE IF YOU WISH, BUT YOU ARE ALSO URGED
TO REPEAT THE EXPERIMENT (TRIAL 2) TO OBTAIN BETTER RESULTS.
    YOUR MEASURED VOLUME IS TOO LOW. CONSIDER THE FOLLOWING POSSIBLE SOURCES OF ERROR:
          DID YOU ENTER WEIGHTS CORRECTLY?
          DID YOU COMPLETELY DECOMPOSE THE KCLO3 IN YOUR SAMPLE?  (DID YOU HEAT LONG ENOUGH)
          DID YOU HAVE AN AIRTIGHT SYSTEM (NO LEAKS!)?
  WHEN YOU NEED ME AGAIN, JUST CALL (CHEMLAB1)!
```

```
      CHEMLAB1

       TYPE IN YOUR NAME - FIRST NAME FOLLOWED BY LAST NAME.  BE SURE TO TYPE IN YOUR NAME
   EXACTLY THE SAME WAY EACH TIME YOU USE THIS PROGRAM.

BENORC



       ANSWER THE NEXT TWO QUESTIONS BY TYPING EITHER 1 OR 2 .

      IS THIS TRIAL 1 OR TRIAL 2 ?
?:
      1
      IS THIS SAMPLE:  (1) PURE KCLO3  OR   (2)  AN UNKNOWN COMPOSITION OF KCLO3 ?
?:
      2
      ENTER THE UNKNOWN SAMPLE DESIGNATION.
A
 PART 2:   ENTER THE FOLLOWING PIECES OF DATA:
(1)   WEIGHT OF OXYGEN COLLECTED.
?:
      .398
(2)   TEMPERATURE, ABSOLUTE.
?:
      300
(3)   BAROMETRIC PRESSURE, CORRECTED
             ( TOTAL GAS PRESSURE )
?:
      741.3
(4)   PRESSURE DUE TO OXYGEN
?:
      714
(5)   VOLUME OF OXYGEN AT STP
?:
      356
(6)   MOLAR VOLUME OF OXYGEN ( IN LITERS )
?:
      23.3

      HAVE YOU MADE ANY ERRORS IN ENTERING YOUR DATA ?
?:
      NO
                                              |-------------MACHINE COMPUTED USING YOUR DATA.
                                  YOUR        |
                                  DATA        *            *--MACHINE COMPUTED THEORETICAL VALUES.
     WEIGHT OF OXYGEN.            0.3980***  0.3980     0.5767
     ABSOLUTE TEMPERATURE.      300.0000   300.1600   300.1600
     CORRECTED BAROMETRIC READING.  741.3000   741.0200   741.0200
     PRESSURE DUE TO OXYGEN.    714.0000   714.3200   714.3200
     VOLUME OF O2 AT STP.       356.0000** 333.5862   403.7104
     MOLAR VOLUME OF O2.         23.3000    26.8200    22.4000

       COMPARISON OF THESE COLUMNS ALLOWS YOU TO CHECK BOTH YOUR CALCULATIONS AND
   THE AGREEMENT BETWEEN YOUR EXPERIMENTAL DATA AND THE THEORETICAL VALUES FOR YOUR SAMPLE.

ERROR MESSAGE INTERPRETATION:
        *    5 - 10 PERCENT ERROR.
       **   10 - 20 PERCENT ERROR.
      ***   20 - 40 PERCENT ERROR.
        $   >40 PERCENT ERROR.
       PLEASE HAND-IN THIS TABLE WITH YOUR FINAL LAB REPORT.
   IF YOUR SAMPLE IS AN UNKNOWN, YOU MAY PROCEED TO PART 3, WHEN READY, BY TYPING CHEMLAB1 AGAIN.
```

```
    CHEMLAB1

     TYPE IN YOUR NAME - FIRST NAME FOLLOWED BY LAST NAME.  BE SURE TO TYPE IN YOUR NAME
  EXACTLY THE SAME WAY EACH TIME YOU USE THIS PROGRAM.

TOMBERR

     ANSWER THE NEXT TWO QUESTIONS BY TYPING EITHER 1 OR 2 .

    IS THIS TRIAL 1 OR TRIAL 2 ?
⎕:
    1
    IS THIS SAMPLE:  [1] PURE KCLO3  OR   [2]  AN UNKNOWN COMPOSITION OF KCLO3 ?
⎕:
    2
    ENTER THE UNKNOWN SAMPLE DESIGNATION.
A
  PART 3:  ENTER THE FOLLOWING DATA:
|1|   MOLES OF OXYGEN.
⎕:
    .165
|2|   MOLES OF POTASSIUM CHLORATE DECOMPOSED.
⎕:
    .110
|3|   WEIGHT OF POTASSIUM CHLORATE IN SAMPLE.
⎕:
    .145
|4|   WEIGHT OF SAMPLE USED.
⎕:
    2.94
|5|   PERCENT OF POTASSIUM CHLORATE IN SAMPLE.
⎕:
    51

    HAVE YOU MADE ANY ERRORS IN ENTERING YOUR DATA ?
⎕:
    YES

    ENTER THE NUMBER OR NUMBERS ( FOUND IN THE | | ) OF THE INPUT STATEMENTS ASSOCIATED WITH
  THE DATA YOU WISH TO CORRECT.  IF MORE THAN ONE NUMBER IS TO BE ENTERED, SEPARATE WITH BLANKS.
⎕:
    3
|3|   WEIGHT OF POTASSIUM CHLORATE IN SAMPLE.
⎕:
    1.45


    HAVE YOU MADE ANY ERRORS IN ENTERING YOUR DATA ?
⎕:
    NO

                                        YOUR          |‾‾‾MACHINE COMPUTED USING YOUR DATA.
                                        DATA          ↓

  MOLES OF OXYGEN EVOLVED.              0.1650     0.0176
  MOLS OF KCLO3 DECOMPOSED.             0.1100     0.0117


  WEIGHT OF KCLO3 IN SAMPLE.            1.4500     1.4350
  WEIGHT OF SAMPLE USED.                2.9400     2.9450
  PERCENT KCLO3 IN SAMPLE.             51.0000    48.7268

  THEORETICAL PERCENT POTASSIUM CHLORATE FOR YOUR SAMPLE IS 50 PERCENT.
  YOU HAVE FINISHED COMPUTER TREATMENT OF DATA FOR THIS SAMPLE, AND MAY NOT ALTER
 ANY OF THE DATA.  PLEASE HAND IN THIS TABLE WITH YOUR REPORT.  IF YOU HAVE MADE ANY
 ERRORS IN YOUR FINAL CALCULATIONS, CORRECT THEM AND INCLUDE THE CORRECTIONS IN
 YOUR LAB REPORT.  YOUR REPORT SHOULD INCLUDE A DISCUSSION OF YOUR EXPERIMENTAL ERROR.
```

```
INSTRUCTOR

[I]:          TUBE AND           TUBE AND        WEIGHT OF         OXYGEN          UNCORRECTED
          CONTENTS BEFORE        RESIDUE        EMPTY TUBE      TEMPERATURE       BAROMETRIC
              HEATING         AFTER HEATING                                      PRESSURE

[II]:        ABSOLUTE           CORRECTED      PRESSURE DUE       VOLUME OF      MOLAR VOLUME
           TEMPERATURE         BAROMETRIC      TO OXYGEN         OXYGEN AT      OF O2 AT STP
                                READING                            STP

[III]:     STUDENT VOLUME           COMPUTED VOLUME
          OF O2:UNCORRECTED        OF O2:UNCORRECTED

STUDENT:  BENORC

[I]:
SAMPLE:B              35.1220         34.7240         32.1770          27.0000         744.3000
SAMPLE:A              35.1220         34.7420         32.1770        0 27.0000         744.3000

[II]:
SAMPLE:B             300.0000        741.3000        714.0000         356.0000          23.3000
SAMPLE:A             300.0000        741.0000        714.0000         400.0000          22.8000

[III]:
SAMPLE:B             390.0000        471.8479
SAMPLE:A             350.0000        471.8479

STUDENT:  TOMDERN

[I]:
SAMPLE:A              35.1220         34.5600         32.1770          27.0000         744.3000
SAMPLE:NONE            0.0000          0.0000          0.0000           0.0000           0.0000

[II]:
SAMPLE:A             300.0000        741.0000        714.3000         401.0000          22.4000
SAMPLE:NONE            0.0000          0.0000          0.0000           0.0000           0.0000

[III]:
SAMPLE:A             468.0000        471.8479
SAMPLE:NONE            0.0000          0.0000
```

Computer-Generated Repeatable Tests in Chemistry

by Franklin Prosser and John W. Moore
Indiana University, Bloomington

The specification to students of performance (behavioral) objectives, as has been pointed out by Jay Young[1], is more commonly (and emphatically) done by means of examinations. Examinations should involve other aspects (self-evaluation, requirement of additional study, opportunity for original thought in applying principles to real situations) which are important in inducing students to learn. In the large-scale classes common in many universities (and to a majority of students in general chemistry) such aspects of testing have been subordinated to the necessity of evaluating a student's progress and comparing him with others. This has come about as a result of lack of time and energy on the part of instructors in large courses, and because of a failure to recognize that such courses differ qualitatively as well as quantitatively from the classes of thirty to fifty taught in the "good old days." The problem is, of course, readily apparent to students, who often fail to find anything worthwhile in our systems for testing, and therefore call for the abolition of all grades. If the only function of our examinations is to order students (to a precision of four significant figures) from best to worst, such calls probably ought to be heeded.

In this paper we present a system of testing which allows the instructor to overcome many of the drawbacks of conventional testing *without significant increases in time, energy or money being spent*, provided the system is used for a four year period. (as with any innovation, a certain activation energy must be overcome initially, but once begun, the system appears to have a negative free energy change.) Although not a form of Computer-Assisted Instruction (CAI) in the defined sense of the term[2], this system is a form of computer-managed (or computer facilitated) instruction[3]. For large-class instruction a computer is required to handle clerical tasks, but there is no on-line interaction between student and computer. This means, of course, that the technological and economic disadvantages of CAI are eliminated. It also means that the option of adopting the system lies with the individual instructor, rather than the university or a federal agency. The necessary computer hardware is available on most campuses and we can provide appropriate software to those who need it. As will be demonstrated later in this paper, many of the advantages of CAI are inherent in our system as well.

The system to be described consists of computer-generated repeatable tests (CGRT). Large numbers of unique but equivalent tests are generated by a computer program which takes stratified random samples from an item pool and prints out questions and answers in a format appropriate to test taking and machine or hand grading. Using this system students may be examined more frequently, encouraging them to keep current and allowing a better opportunity for self-evaluation. Students get immediate (within forty five minutes) feedback since answer keys (including suggestions for remedial study) are provided as soon as the test is over. Most importantly, the exams, because they are unique, may be given repeatably. This allows the student to take a test, discover some of the performance objectives he had not thought of before,study or review to increase his mastery, and retake the test. These exams have real pedagogical value, and much of the trauma associated with testing large classes has been eliminated.

The Computer Generated Repeatable Testing process typically consists of four steps: (1) developing pools of test items, (2) producing tests, (3) administering the tests, and (4) scoring the tests. The second and fourth steps are managed by computer, while the execution of the first and third steps is strongly influenced by the computerized nature of the process.

For each exam, the course instructor develops a pool of items (test questions) which forms the data base from which tests are prepared. This is a rather formidable step. Our experience indicates that one should have at least six to ten items in the pool for each question on an exam to assure adequate variation on the individual tests. An instructor planning to give eight exams of twenty questions each should construct about fifteen hundred individual items for his course. This work, although it has the advantage of being more familiar to most instructors than programming CAI, is every bit as tedious and time consuming as it sounds. It should be done prior to the first semester in which repeatable testing is to be used in the course. Fortunately, the item pools, once developed, are rather permanent, especially for the basic college undergraduate courses that are the most likely candidates for this computerized testing scheme. Only relatively minor alterations to the item pools are needed to accommodate other instructors, changes of texts, etc., that may occur in subsequent semesters. Further, textbook publishers often have compendiums of test questions for their popular texts. Reusing test questions semester after semester, or even making the entire item pool available to students, is not a disadvantage under our procedure, and in fact is likely to be distinctly advantageous, since it is merely a specification of behavioral objectives.

Since items will ultimately appear on computer generated tests, the form of the items must conform to the requirements of present computer printing technology. Normally, items may consist of upper-case letters, numbers, and the usual special characters available on modern high-speed line printers. Diagrams, pictures and other graphic aids cannot usually be printed directly.

although the instructor may easily include these by providing the student with a supplementary sheet of diagrams to accompany the tests. Our experience has been that such "handouts" are quite adequate for presentation of structural formulas and other chemical information not readily printed.

If the tests are to be graded manually, technology imposes no limitation on the structure of the answer to an item. The test questions may elicit objective or subjective responses from the student. On the other hand, if the instructor wishes to use mechanical grading techniques, he must provide for a single-character response for each item, because of restrictions imposed by the optical mark sense form readers usually available in universities. While this requirement may appear to be a severe limitation, it in fact allows considerable freedom in the form of objective test items. True-false and multiple choice items call for single character responses. Key-word, fill-in, and other forms resulting in a definite numeric or symbolic answer may easily be reduced to a single character response using the following convention: In such a question the form of the answer is indicated by a series of dots which includes one asterisk. The student will construct his symbolic or numeric answer to the question, and will record as his response on his mark sense form the single character selected by the position of the asterisk in the string of dots. For example, ..*. means code the third letter or digit of the answer, *... means the first letter or digit, and so forth. Students describe such alphabetically or numerically coded items as being hard but fair. The student cannot answer such an item unless he has mastered the basic concepts and vocabulary. Recall is emphasized; simple recognition is subordinated.

In addition to the question part of an item, which the student sees when he takes a test, each item also has an answer part to allow machine grading and to provide information to the student after testing. The answer part of an item may contain, in addition to an answer character, any relevant information, such as the full symbolic or numeric answer, textbook page references, and other diagnostic aids for the student.

After the instructor has developed a section of his test item pool, he will have it punched onto punch cards or entered into an appropriate editable data file system. To facilitate the selecion of items for an individual test and to maintain order among the large item pools the instructor classifies his items into sets, the items within a set are given distinct unit numbers, and cards or lines for the question part and the answer part of each item are numbered serially. Usually a set will consist of those items that test similar material.

The individualized tests are generated on a digital computer using a computer program GENERATOR. This program reads the item pool for a particular exam, checks the input data for proper sequencing and correct format, reads information describing the tests to be generated (number of tests, number of questions per test, etc.), generates and prints the individual test, and punches a small answer summary deck for use in mechanized grading. Each test is individually numbered and has questions on the left part of the line printer page and answers on the right. The item identification numbers for each question appear in the answer part for reference. The instructor will of course separate the answer part from the question part prior to giving a test to the student.

The computer program selects items for a test by randomly choosing an item from each set. The order of choosing sets is also randomized. No item is used more than once per test. The digital computer is vital to test productions, since the random item selection, formatting, and printing of large numbers of individualized tests is beyond the capacity of nonautomated operations. The instructor may also assign weights (point values) to sets of items, thus allowing him to emphasize particular topics or award points based on the difficulty of items.

The computer time required to generate the tests is very small; the time required to print tests is, however, substantial. Typical times on the Indiana University CDC 3600 computer system are about four minutes of computer time (of which about 20 seconds are for item selection) to generate 1,000 three-page tests, and about three hours of printer time to print them. As we show later, the total cost per test is about 5 cents.

The instructor decides for himself how and when to test his students. He may give tests in class or at other scheduled times; or, more flexibly, he may allow his students to choose their own times for testing. A combination of in-class testing preceded or followed by opportunities for student-scheduled testing appears to be useful. Such options depend on the instructor's preference and the availability of testing room space and personnel.

A student taking a test usually obtains an individualized test (with answer part removed) and a mark sense form and special pencil. He takes a seat in the testing area and immediately enters on his mark sense from his student identification number (social security number or other agreed-upon identifier), the exam number, and his individual test number. The student then marks his answer on his test, and for each question enters the appropriate single-character response on his mark sense form. After completing a test the student exchanges his mark sense form for the answer part of his individual test. The mark sense form is kept by the proctor for later grading. The student, having the correct answers in hand, can immediately determine his errors, and is stimulated to improve his knowledge of weak areas.

The instructor or his assistant will, whenever convenient, have the information on the mark sense forms transformed to punch cards on an optical mark sense form reader. This step is required to obtain a form of input acceptable to the typical academic computing facilitity; one can bypass this step if optical mark sense form reading equipment is attached directly to his institution's computing equipment.

Scoring of the student responses for an exam is done by computer using program GRADER. Input to this program is the answer summary deck punched by program GENERATOR when the tests were prepared, and the student response cards derived from the mark sense forms. Output of this program is a roster of student ID's and test scores and a punch card deck of the high score for each student for this exam.

As a followup of test scoring, we are developing an item analysis procedure for CGRT. Since the item pools tend to be reused many times, such an item analysis will aid in the detection of defective test items and will assist the instructor in polishing his item pool.

The test producing program GENERATOR and the scoring program GRADER are written in Fortran. Virtually all acedemic computing centers have well-maintained Fortran compilers that produce a fairly good quality object code. We have several versions of the CGRT programs: well-documented ANSI Fortran versions designed to run on all commonly-available computers, and specialized versions of GENERATOR for the CDC 3600 and for the CDC 6600. The specialized versions utilize CDC extensions of ANSI Fortran to decrease the execution time dramatically by by-passing the repetitive processing of format statements during test printing. Since GENERATOR is completely output-bound, we anticipate that many potential users of the ANSI Fortran version would wish to discuss modification of the program with their systems people to take advantage of local extensions to Fortran output facilities.

At first glance a procedure that uses a computer for test preparation and for printing of individualized tests appears economically unsound. This is very definitely not the case. In Tables I and II, which are cost analyses for the preparation, printing, and scoring of 1000 three-page tests, we have attempted to itemize expenses in a similar manner for both CGRT and the conventional method. Therefore, the cost of a computer line printer and associated equipment has been treated as a separate entry, rather than included in general computer charges. We have assumed that such expenses as the initial keypunching of item pools are distributed over four semesters.

The analyses show that both CGRT and conventionally prepared tests cost about 5 cents per test (exclusive of the instructor's time required to generate questions). While the estimate for conventional exams is fairly accurate, changing some of the assumptions in the CGRT analysis may alter the estimate by perhaps up to two cents per test. Also, under repeatable testing, students tend to take more than one repeatable test over each examination unit. In any event, the cost of repeatable tests is in the same range as conventional tests. More important, the expenses of the CGRT process are a very minor item in the cost of educating the student, amounting to $.50 to $1.50 per student per course. This is inexpensive education!

The results of an attitudinal survey of about four hundred students in chemistry C105 at Indiana University during the fall semester, 1970, are reported as percentages in Table III. The survey was administered shortly after the Christmas break, but prior to the last repeatable exam and the non-repeatable final exam.

Some specifics as to the operation of CGRT in chemistry 105 are in order before the survey results are discussed. Seven exams were given during the fifteen week semester. In each case exams were made available at the beginning of a two-week period. Students were allowed up to three "takes" of each exam, with only the highest score recorded for the final grade. On the deadline for taking each exam the usual repeatable test forms were administered in lecture class. A few students took make-up exams after this date, but this was not common. Computer grading was necessary because of the large number of exams being given, so answers were recorded as single characters on mark-sense forms. In cases where a question was ambiguous or incorrect answers were suspected, students used a "complaint sheet" to request additional credit. Structural formulas and other like information were presented by means of "handouts" which were coded to correspond with certain sets of questions.

With regard to item number one on the survey, the pressure involved in large-scale testing is evident to anyone who has walked into a room with 400 students waiting to take a test. The tension is so high that it often given seasoned teachers some qualms. Its effect on students is, of course, much greater. Moreover, most of the students in such large-scale classes will not often be faced with making decisions under similar conditions of stress, except while taking college exams. We feel that removal of a portion of this tension is beneficial in that it improves students' attitudes as well as providing a more realistic evaluation of their performance and ability.

More than three quarters of the students polled thought that repeatable exams helped them to learn more chemistry. Only six per cent felt that the exam system was of doubtful or no pedagogical value. This is somewhat less than the number who had similar feelings about the textbook. While student opinions cannot be substituted for experimental data on the amount of actual learning taking place (which we are in the process of obtaining) it seems quite clear that most students are very happy with the CGRT system. This is also apparent in the third statement where nearly eighty per cent of the class agreed that the system offered an opportunity to demonstrate their "real understanding"of the course material.

The fourth survey statement was included because students who were the initial guinea pigs for the CGRT system during the 1970 summer session complained that they had too much incentive to study and might therefore neglect their other courses in favor of chemistry during a normal semester. It is encouraging to see that during the fall semester most students did not feel that unreasonable amounts of work were required by CGRT. Indeed, it may be possible to include more material in a course without student complaints when CGRT are used. That is, more efficient learning is taking place.

The next two statements (5 and 6) were included because cheating has always been a problem in large scale testing situations. While the incentive to cheat is less with CGRT, it still remains. The forms which cheating can take, however, are different because of the individuality of the test forms. In our experience proper administration of the tests can eliminate all forms of cheating (or at least all that have come to mind or been tried so far) except blatant collusion between two students in the exam room. This latter can be detected easily by alert proctors.

Statements 7 and 8 in Table III indicate that most students find the repeatable test forms useful as study guides (or behavioral objectives). They also have the valuable experience of consulting with their peers when they have a problem. They are much more likely to study together because the CGRT system pits everyone against the computer, rather than every man for himself against everyone else. Once a student has achieved an "A" he has an "A" even if it turns out to be the class average. The ominous "curve" is absent because performance levels are set by the exam questions. If these levels are met the student gets credit independent of whether others meet the same levels or not.

Statement 9 and 10 reflect two other aspects of the CGRT system. In a number of cases information was tested which had not been presented in lecture, discussion, laboratory or the textbook. That is, a few (relatively minor) *portions of the course were taught by means of repeatable tests.* There are many aspects of chemistry which lend themselves to this approach. For example, simple stoichiometry problems, equation balancing, chemical symbols, valence, and some aspects of descriptive chemistry require considerable (very unpopular) rote learning. One often wishes to bypass such topics in lectures, and our experience indicates that this can be done using CGRT. Students faced with such problems on exams simply find someone who can explain to them how a certain type of problem should be approached. Having learned this, they solve it correctly on the second or third try.

Statement 10 indicates that with careful framing of questions a test restricted to single-character responses need not be picayune in its approach. Of course, as with any type of behavioral objective, there is often a tendency to emphasize details because they are more readily specified. Nevertheless only about one-third of the C105 students felt that the major emphasis was on details as opposed to general principles. It must be reiterated, however, that the CGRT system is only as good as the questions that are written for the item pool. It does have the advantage that poor questions can be weeded out, but a large number of good, imaginatively written questions must be available. These questions specify, to a much greater degree than in the case of ordinary tests, what students will study and learn.

The last two statements in the attitudinal survey simple reemphasize the success of the entire course. Although other factors than the exam system obviously affected these student attitudes, the CGRT system was responsible for much of this success.

Computer Generated Repeatable Testing works. It has been used in numerous courses for nearly three years at Indiana University, and it is also in use at the University of Nebraska, Illinois Institute of Technology, Indiana University-Purdue University at Indianapolis, and other places. The method has been enthusiastically used by instructors of undergraduate courses in such varied disciplines as psychology, chemistry, computer science, economics, English, speech therapy, home economics, accounting, and education.

In general, students have been highly satisfied with the repeatable testing method. Their mood is one of alertness rather than anxiety. They are relaxed during examinations, and their morale is good. The undergraduate counselling units of Indiana University have received numerous student comments favorable to CGRT.

An unexpected result in some of the CGRT courses has been the students' excellent performance on technical material not discussed in class. Repeatable examinations appear to provide a stimulus and a way to master material typically neglected by students in conventional courses. Although we have only a little data taken under properly controlled conditions, indications from several common achievement tests given at Indiana University are that overall student achievement in repeatably tested sections is higher than in conventionally tested sections of the same course.[4] All available evidence suggests that a system of frequent and repeatable examinations provide an excellent atmosphere for scholarly activities of beginning students.

We hope that many readers will wish to try CGRT or suggest its use to their non-computer-oriented colleagues. The computer programs and ample documentation are available from Franklin Prosser.

References

1) Jay A. Young, Abstract No. ACSC 5, 161st ACS National Meeting, Los Angeles, California, March 29, 1971.

2) P. Suppes and M. Morningstar, Computer-Assisted Instruction, Science, Vol. 166, Pages 343-350, 1969.

3. W. C. Cooley and R. Glaser, The Computer and Individualized Instruction, Science, Vol. 166, Pages 574-582, 1969.

4) D. Jensen and F. Prosser, Computer-Generated, Repeatable Examinations and Large Class Instruction, Presented at Midwestern Psychological Association Meeting, Chicago, Illinois, 1969.

Table I. Cost Analysis of CGRT.[a]

ITEM	COST
Punched cards for item pools (one-time expense)[b]	$ .40
Punch cards for student responses: 1000 cards	1.00
Printer paper: 3000 sheets	8.30
Mark sense forms: 1000 forms	8.80
Keypunching services for item pool punching (one-time expense)[b]	6.50
Computer charges for test production and grading: about 5 minutes at $200 per hour[c]	16.70
High speed line printer and controller rental and maintenance: at $1800 per month[d]	6.00 to $12.00
Total expenses =	$47.70 to $53.70

AVERAGE COST PER TEST: 4.8 cents to 5.4 cents

[a] for 1000 three-page tests

[b] prorated over four semesters

[c] Indiana University CDC 3600 system

[d] CDC 512 printer system

Table II. Cost Analysis of Conventionally Prepared Tests.[a]

ITEM	COST
Paper: 3000 sheets	$6.00
Mark sense forms: 1000 forms	8.80
Punch cards for student responses: 1000 cards	1.00
Clerical services at $4.00 per hours:[b]	
Typing: 1½ hours	6.00
Multilithing: 2 hours	8.00
Collating and stapling: 4½ hours	18.00
Computer charges for grading: about 1 minute at $200 per hour[c]	3.30
Total expenses	$51.10

AVERAGE COST PER TEST: 5.1 cents

[a] for 1000 three-page tests

[b] estimates supplied by Indiana University Chemistry Department

[c] Indiana University CDC 3600 system

Table III. Results of Attitudinal Survey

1. To what degree do you feel that repeatable exams remove the pressure involved in achieving a good grade?
Very greatly 57.00% Somewhat 13% Actually increase
Considerably 25.50% Not at all 3.00% pressure 1.50%

2. Do you feel that a repeatable exam, of itself, has helped you actually learn more material?
Very definitely 77.50% Don't really Doubtful 2.00%
Probably 16.00% know 2.00% Definitely not 2.00%

3. The repeatable exam system gives me an opportunity to demonstrate my real understanding of the course material.
Agree strongly 38.00% Disagree moderately 8.75%
Agree moderately 41.00% Disagree strongly 4.00%

4. The repeatable exam system is just another device to get students to do unreasonable amounts of work for the credit given.
Agree strongly 1.00% Disagree moderately 15.00%
Agree moderately 3.00% Disagree strongly 77.00%

5. A student should feel less compulsion to "beat the system" when repeatable exams are used.
Agree strongly 45.50% Disagree moderately 7.50%
Agree moderately 32.25% Disagree strongly 3.00%

6. Do you think it is possible to "beat the system" when computer-generated repeatable exams are employed?
Very definitely 8.00% Wouldn't know 37.00 Very definite-
Probably 31.00% Probably not 19.00 ly not 4.00%

7. Do you find that keeping the exams and using the answer key serve as a good guide to remedial study?
Very great help 86.00% No help Only confuses me 0%
Some help 12.00% 2.00% Worse than nothing 0%

8. Do you find that discussing one another's exams increases your mastery as well as bringing about a change in your score?
Very much 71.00% Not at all 4.00% Worse than no
Somewhat 24.00% Only confuses me 1.00% discussion 0%

9. The exams assume too much chemical information not presented in C105.
Agree strongly 5.00% Disagree moderately 38.50%
Agree moderately 28.25% Disagree strongly 19.00%
No position 8.50%

10. The emphasis in C105 is on details rather than matters of general importance in chemistry.

Agree strongly	7.00%	Disagree moderately	39.00%
Agree moderately	25.50%	Disagree strongly	19.00%
No position	10.00%		

11. Since taking C105, my interest is chemistry has:

Increased greatly	29.00%	Remained unchanged	21.50%
Increased slightly	42.25%	Decreased	6.25%
Terminated	1.0%		

12. The C105 course as an intellectual challenge is:

Very great	12.00%	Average	Below average	4.00%
Above average	58.50%	24.50%	Far below average	1.00%

The Effect of Computers on the Teaching of Physical Chemistry

By Morris Bader
Moravian College

Physical Chemistry is a subject ideally suited for interaction with a computer. The course itself is an investigation into the nature of things with emphasis on the laws of nature which lend themselves to a mathematical analysis of natural phenomena. So, for example, the Clausius-Chapeyron equation tells us how the vapor pressure of a liquid varies with temperature; the Van Der Waal's equation tells us how real gases behave under extremes of pressure and temperature, and even such esoteric equations as that of Schrodinger tell us how electrons behave in atoms and molecules.

In the past, we college professors thought we were teaching good chemistry by having the student plug numbers into equations ad infinitum. Today, more is needed and more is now possible. It still behooves a student to understand mathematics but we would also like him to understand the underlying concepts. Here is where the computer plays a most important role. The student, and advanced students at that, who may never know how to solve Schrodinger's equation for the electron density in a molecule might still understand the concepts involved if he could be led to ask those questions which explore areas of such mathematical complexity that it would be senseless to pursue that route. The computer program HMOLC[1] written at Moravian College for the IBM 1130 computer allows the student to ask the computer to construct a plot of the electron density around 2 atoms at any distance of separation, for any type of electron configuration. That picture is worth a thousand lectures. On a course devoted to principles one must always be on guard to balance the heavy theoretical with the manifestly practical.

The laboratory is one place where pedagogical breakthroughs can be made with a computer. First, routine calculations can be totally eliminated. The impact of this on the student is amazing. It encourages the student to become more precise in his collection of experimental data because it becomes rather obvious when final results are computed where he has been lackadaisical. The student is also encouraged to take more data since he knows that the tedium of calculations is no longer necessary. For example, the program DERIV will accept any type of analytical titration data such as pH vs mls, and automatically plots the data, and then plots both the first and second derivative. All instructors know it takes a student literally hours to plot one run while the computer does this in seconds. Before the computer, the tedium of the plots completely counteracted the intent of the learning situation, that is, a student learned never to do a pH titration ever again. It is not necessary to mention that DERIV can expand the region of the equivalence point to offer mathematical accuracy more than ten times the experimental limitation. So, not only is tedium removed, but all necessary scientific ingredients also improve. Occasionally, workup of lab data is so rugged an instructor recoils from attempting such an experiment. An example of this would be the experiment in partial molal volumes of liquid mixtures. Weights must be corrected for air buoyancy, a pyknometer (liquid density device) must be calibrated, densities and molecular weights must be converted to concentration units, and a frightening list of other calculations follow. The program MOLRV does all these calculations and completes the job with the final plot of the molal volumes. While it is expected that a student understand the calculations and even show samples in his lab report, the computer emphasizes the scientific aspects of the laboratory.

Finally, we have the area of scientific research. Can a person in a small college with a small computer hope to compete with the giant universities with their giant machines? The answer is a definite yes!

At Moravian College we have been able to do fundamental research in the molecular structure of diatomic molecules. The program MOSTR [2, 3] accepts data from the infrared absorbtion spectrum of a diatomic molecule in the gas phase and computes a number of molecular parameters in the ground, first, and second excited states including the anharmonicity constant. This program has been included in the review article [4] on

"Energy States of Molecules" as the state of the art. Such a commendation acts as an enormous plus in an instructor's bag of tricks. A set of 4 programs with research capabilities have also been published by this author[1]. These include an NMR Simulation, a Huckel Molecular Orbital Calculation, and Atomic and Molecular electron density contour diagrams, all written for the small computer.

To summarize, physical chemistry can no longer be taught without the computer as an integral part of the subject. Hopefully, with imagination and perseverence other disciplines might find themselves in a similar excited state.

References

1. "Computer Programs in Undergraduate Chemistry", J. Chem. Ed., 48, 175, (1971).
2. "A Computerized Physical Chemistry Experiment", J. Chem. Ed., 46, 209, (1969).
3. Quantum Chemistry Program Exchange, Program No. 164, "MOSTR - Molecular Parameters of Diatomic Molecules", Indiana University, Bloomington, Indiana, Jan., (1971).
4. "Energy States of Molecules", J. Chem. Ed., 47, 2, (1970).

9-44

A HYBRID APPROACH TO COMPUTER ASSISTED INSTRUCTION

By

F. D. Tabbutt
Reed College

Portland, Oregon 97202

Although analog and digital approaches to assisted instruction have complementary qualities, their presentations are not always complimentary to each other. The analog computer users listen to the new and old methods of calculating and displaying digital solutions to the damped oscillator with a certain superiority that stems from the knowledge that they can do it so much more quickly and inexpensively on an analog computer. The digital computer user has a similar superiority when listening to the papers on analog CAI, for he knows that although the analog medium is highly interactive, the choice of problems is extremely limited. In fact, in this day of sophisticated electronics, the stone age image which the analog patch panel presents whether it be time-shared or not, is not likely to entice very many new members into the club.

These observations are not new. Many have seen and probably experienced them. Most people involved in CAI have recognized the chasm and the sometimes wasteful competition between the two basic types of computer instruction. Most have also been aware of hybrid systems, which if properly designed can take advantage of the superior qualities of both systems. This paper is a report on progress along the next obvious step — time sharing a hybrid computer for instructional purposes.

General Objectives and Design of the System

The primary objective of this instructional system has been interactive graphic flexibility in dealing with mathematical topics. To achieve this end the system was to minimize the need for language format and programming on the part of the user, and to maximize speed of response and graphic quality. Moreover, it was anticipated that the bulk of the topics to be treated could be represented by differential equations.

Theoretically, at least, the design of a hybrid system to accomplish this is reasonably straightforward. First, the duty of solving and plotting the differential equations is assigned to the analog computer — the obvious choice for reasons of speed and graphic quality. To the digital computer is assigned the role of controlling the time-sharing, i.e., servicing the time-shared terminals. However, to deal with any differential equations that any user might specify, it will be necessary for each user to have the analog computer patched to his specifications for his problem. Speed limitations alone rule out patch panels at terminals; the alternative is an automatic patch cord mechanism or autopatch. Autopatch can be achieved with a large number of high speed switches which are each under control of the digital computer. At this stage in the design, then, a user would have to write out the analog program for his problem, look up the switch addresses to implement it, and then instruct the digital computer to close them. This is messy and hardly meets the goal of freedom from programming and language format. Fortunately, there is another way.

Those who are familiar with the general(1) or bootstrap method of solving differential equations with an analog computer must recognize the mechanical nature of the approach: 1) isolate the highest derivative, 2) integrate and manipulate until all terms equal to highest derivative have been generated, 3) close loop, 4) set initial conditions. Given some specific complement of analog components and the switch paths between them it would seem quite feasible to have the digital computer step through an algorithm to determine what is to be connected to what and then, with the control it has over the switches, to make the proper connections. In summary, then, if we have the autoprogramming capability just described, and the autopatch capability described earlier, it would then become a simple task to have the digital computer recognize an alphanumeric statement of a differential equation and solve it on the analog computer.

However, there yet remain two hurdles to be cleared before this system is, theoretically at least, complete. Parameters must be adjustable for each terminal. There are a number of ways of doing this. Potentiometers can be arranged at each terminal and switched in and out as dictated by the digital computer controller.

Alternatively, servo set pots at the computer could be controlled by the terminal. Yet, for operation at speeds which the modern analog computer are capable, e.g., 0.2 msec initial condition, 1.0 msec operate, neither method is satisfactory. The phase shifts in remote pots attendant with frequencies to be expected at these speeds (~10 KHz) void the former possibility. The sluggish response time at servo set pots eliminates them. Since autopatch requires so many coefficient devices, we can only realistically consider time-sharing coefficient devices which can be set in less than 0.1 msec. Such coefficient devices do exist: they are digital to analog converters capable of multiplying an analog variable. We will call them MDACs for short.

Finally, there is the annoying, albeit important, requirement to communicate user desires to the digital computer. A keyboard would be the simplest way, but a character generator which would enable a display of the statement of the equations is required. A stroke generator with analog output would be preferable to be compatible with the output from the analog computer.

An overall schematic of what has just been described is shown in Figure 1. The hardware and software details of this system are described in the next section. Following that is a section which summarizes the system and points to some uses.

The Construction of the System

Autopatch

Several methods of autopatching an analog computer have been proposed.(2-5) More sophisticated approaches have been a telephone-like trunking system to reduce the number of switches involved. Without trunking, the number is simply the product of the number of inputs (I) and outputs. Since in analog computing there are about twice as many outputs as inputs about $2I^2$ switches are required. Sophisticated trunking can lower this at best to $I^{1.5}$. However, if one deals with a small number of analog components, say 15, then if all outputs can be connected to all inputs only 450 switches are required. Furthermore, if coefficient devices (MDACs) are dedicated to each output so that they need not be counted as a separate component, the 15 components can represent a respectable computing ability. In fact, with the 18 components shown in Figure 2, it was felt that we would have a reasonable start at autopatch with the capability of solving the usual range of equations to be encountered at the freshman or sophomore level. Furthermore, the job of the programmer designing the autoprogramming feature was simplified by avoiding the added complication presented by trunk networks between output and inputs. Consequently, as shown in Figure 2, all outputs had direct connections available to all inputs which is the method used by Howe, et. al.(5) Thus, our initial autopatch system contains five integrators, three multipliers, one divider and three summers, and thirty MDACs connected together by 540 switches. A thirty-first MDAC sets initial conditions.

A great deal of effort went into the design of the analog switch. It had to have a low phase shift (<.056° at 10 KHz) for acceptable accuracy in high speed analog computation. Furthermore, when the switch was open, less than .1% could feed through at 10 KHz. The switch, which was finally used, is shown in Figure 3. It has a primary and a secondary memory in the quad latches that control each switch. Consequently, while one problem is running, the digital computer can be setting the quad latches for the next problem and when the time comes the computer can set all 540 switches in a single command. The switch and the series resistance are adjusted to an accuracy of 0.1% so that all outputs go through switches directly to summing junctions of input amplifiers. The summing capability of such a scheme guarantees that we have, in fact, a summing capability which is greater than the three summers would lead one to believe. The switches are assembled in groups of sixteen on a single printed circuit card. These sixteen-switch cards are inserted in a card cage along with communication cards which control them and the whole assembly is then mounted on the front of the patch panel which is dedicated to Autopatch.

With the advent of the MDAC, the last moving part in the analog computer is replaced. That is, we can now take full advantage of the tremendous analog speed. The MDACs used in this system are simply an array of precision resistors with solid state switches. The ones used in this system are bipolar and have 12 bit accuracy.

They are double buffered, i.e., they have a primary and secondary storage, so that like the switches they can be prepared for the next problem while the present one is running and then all set with a single command.

We, therefore, have an autopatch system in which a problem can be patched and the coefficients set in two commands or about 20 μ seconds with a Nova digital computer.

Character Generator

The character generator is basically an analog function generator for the X and Y axis. As shown in Figure 4a, inputs of +, 0 or - reference voltages can be switched into an integrator during the period of a clock pulse to generate a negative ramp, a horizontal bar or a positive ramp respectively. A sequence of 32 clock pulses activates in turn 32 different control lines and then repeats. The 32 control lines are connected to a gated decoder for X and Y and the X *vs* t and Y *vs* t patterns shown in Figure 4b are obtained. An X *vs* Y plot produces the image shown in Figure 4c. The intensity or Z axis sequence will determine which stroke is unblanked and thus what the character will look like. The z axis sequence is produced as follows. The 32 control lines mentioned above also each go to an array of 32 gates which are "Anded" with the output from a read-only-memory (diode board). The read-only-memory is activated by a seven bit ASCII string from the digital computer. Two eight bit strings are required to set the DACs for the X-Y position of the character. Four more bits control the four switches for input/output impedances of the last operational amplifier. This enables the digital computer to select 16 character sizes and the spacing will be automatically adjusted. The digital computer need only devote the time required to send two command words (~10 μ seconds for the Nova) to produce a character. The speed with which the character is generated then depends on the clock rate. Characters can be generated as quickly as 32 microseconds (1 MHz rate) although for a storage oscilloscope display 0.64 milliseconds (50 KHz rate) is the fastest rate possible for storage. At present one character generator services all terminals.

Interface

The details of the keyboards - CRT - analog computer (AD/5) - character generator - digital computer (Nova) and analog switches interfacing are summarized in Figure 5. Two interfaces (I/F) are mounted on an accessory card inside the Nova computer. The 32 bit I/F effectively controls what the user will see while the 12 bit I/F brings in the data from the keyboards. In addition to the 27 of the 32 bits identified in the character generator section, four are for the address of the terminal and one is an erase pulse for the storage oscilloscope. The character generator has an analog and a logic switch system to determine whether the X, Y, and Intensity signals come from the AD/5 or the character generator. A run command from the Nova sets the switches for character generator outputs, otherwise AD/5 outputs go through. After the Erase command has been given nearly 0.5 second is required before the storage oscilloscope can be used again. Busy is an incoming bus from a flip flop in the oscilloscope which is high only during this interval. The Nova can then test the Busy bus periodically rather than use computer time to count the 0.5 seconds. The selection of the terminal with access to the Data bus occurs as follows. When a user has a character loaded into the keyboard register, then the next time the 50 KC Line Available bus is high that terminal will stop the Line Available signal at the Nova and then clock out the seven bits on the Data bus until the software sets Done at which time the Line Available is activated again.

The interface between the Nova and the AD/5 has a powerful communication linkage. It is this interface which sets the MDACs. Moreover, a cable containing 16 data lines and 6 command lines connects this interface to the array of analog switches. The switches are set with a serial bit string at a cost of one bit/switch. Thus, once initiated, each digital word sets 16 switches.

Text and an unexpected audio dividend

While the terminal has the graphic capability to display text, at a cost of 2 digital words/alphanumeric character, the 8K memory of the Nova will allow only a limited amount of text to be stored in core. Yet it would be helpful if an instructor could compose a CAI lesson which would consist of passages of text alternating with computational computer graphics under control of the user. However, as the library of these lessons increased in number it was clear that they could not all be available from core.

An alternative was to record the text on magnetic tape cassettes. Each terminal would then be provided with a playback device. When the cassette recorder is connected to the keyboard and playback initiated, the digital information coming from the tape appears to the Nova as a very rapid typist. The magnetic cassette has large storage capability so that any number of lessons could be maintained in a library. All a student need do is pick out the cassette with the desired lesson and fast forward to the counter reading where the lesson begins.

The cassette player is a commercial device which has dual channels for stereo use. One channel is committed to digital information which still leaves the other for audio. This unexpected dividend makes it possible for the CAI lesson to have graphic and audio presentation simultaneously for little additional cost.

Hardcopy

A useful, although sometimes expensive, feature of a graphics terminal is the ability to give hard copy. This can be done in a straightforward manner with the circuit shown in Figure 6. The circuit samples a repeating analog signal and slowly moves the sampling window across it without ever affecting the speed of the system it is sampling. By doing this tc both X and Y coordinates it is possible for an X-Y recorder to record in several seconds a graph which has been calculated in 1 msec several thousand times. The hardware for this sampling system is available on the AD/5 and can be permanently patched on the Autopatch panel. Presently there is a single sampling bus on the system and a single X-Y recorder that can be used at any terminal.

Software

An important, if not crucial, aspect of writing the software for the system was the ability to write it on cards to simplify editing. This was particularly true for the Autoprogramming section. However, our Nova only had a teletype-paper tape input. Accordingly, some time was spent initially writing an emulator for the Nova on the IBM 1130 which resides at the Reed College Computation Center. Thus virtually all programs were written and tested on the 1130. Only then were the cards converted to paper tape and loaded into the Nova.

An overview of the software is given in Figure 7. There are effectively two different functions being performed by the CPU with buffers as a link.

Consider the following example. Suppose that Control is having Autoprogram work on someone's problem when someone at another terminal depresses a keyboard key. This causes a Control Interrupt due to a character coming in. This is the highest priority so Control pauses while the character is entered into the 200 word

Character buffer allocated for that terminal; then Control resumes its job. When the character stream for entering the problem by the user is complete, he depresses a control key followed by A. The next time that terminal is serviced the characters in the Character buffer will undergo an Autoprogram. When the Autoprogram is completed the Character buffer now contains the autopatch switch settings and the buffer is transferred to a 150 word Plot buffer for that terminal. Control then requests the terminal user to assign values for each coefficient identified in the problem. These inputs are identified by the control key followed by C and are entered in the Character buffer. When Coefficient sees that all coefficients have been given a value it stops asking, transfers the buffer to the Plot buffer and awaits a run command. The user initiates this by depressing the control key followed by R. The next time Control is interrupted by the AD/5 signifying that it is ready and it is the turn for this terminal, the Plot buffer will be transferred to the AD/5 and then a command given to computer. The run command can also cause Reset to clear both buffers for that terminal.

Summary

A hybrid CAI system has been described which has the interactive capacity of a digital computer and the speed of an analog computer with an audio capability thrown in. It would appear that such a system offers great potential. It can service a large number of users solving any number of differential equations at a speed which has not been achieved before. This hybrid time-share system is a small step in a direction which could lead to a fruitful partnership of digital and analog techniques.

It is too early to tell at this writing what combination of text, graphics, and audio is best. It is too early to say how effective this system will be as an instructional device. Specific examples of the use of the system will be given at the presentation of this paper.

Acknowledgements

The results reported here were made possible through the generous support of the National Science Foundation (GJ-326). It is also a pleasure to acknowledge the important contributions of Peter Langston, Michael Penk and Mark Roberts in software and of Willard West in hardware assistance.

Bibliography

1. L. Levine, *Methods for Solving Engineering Problems*, McGraw-Hill, 1964; p. 45.
2. D. A. Starr and J. J. Jonsson, "The Design of an Automatic Patching System," Simulation, *10*, 281 (1969).
3. G. Hannauer, "Automatic Patching for Analog and Hybrid Computers," Simulation, *12*, 219 (1969).
4. T. J. Gracon and J. C. Strauss, "A Decision Procedure for Selecting Among Proposed Automatic Analog Computer Patching Systems," Simulation, *13*, 133 (1969).
5. R. [illegible]. Howe, R. A. Moran and T. D. Burge, "Time-Sharing of Hybrid Computers Using Electronic Patching," Simulation, *15*, 105 (1970).

Overview of a Time-Shared Hybrid System

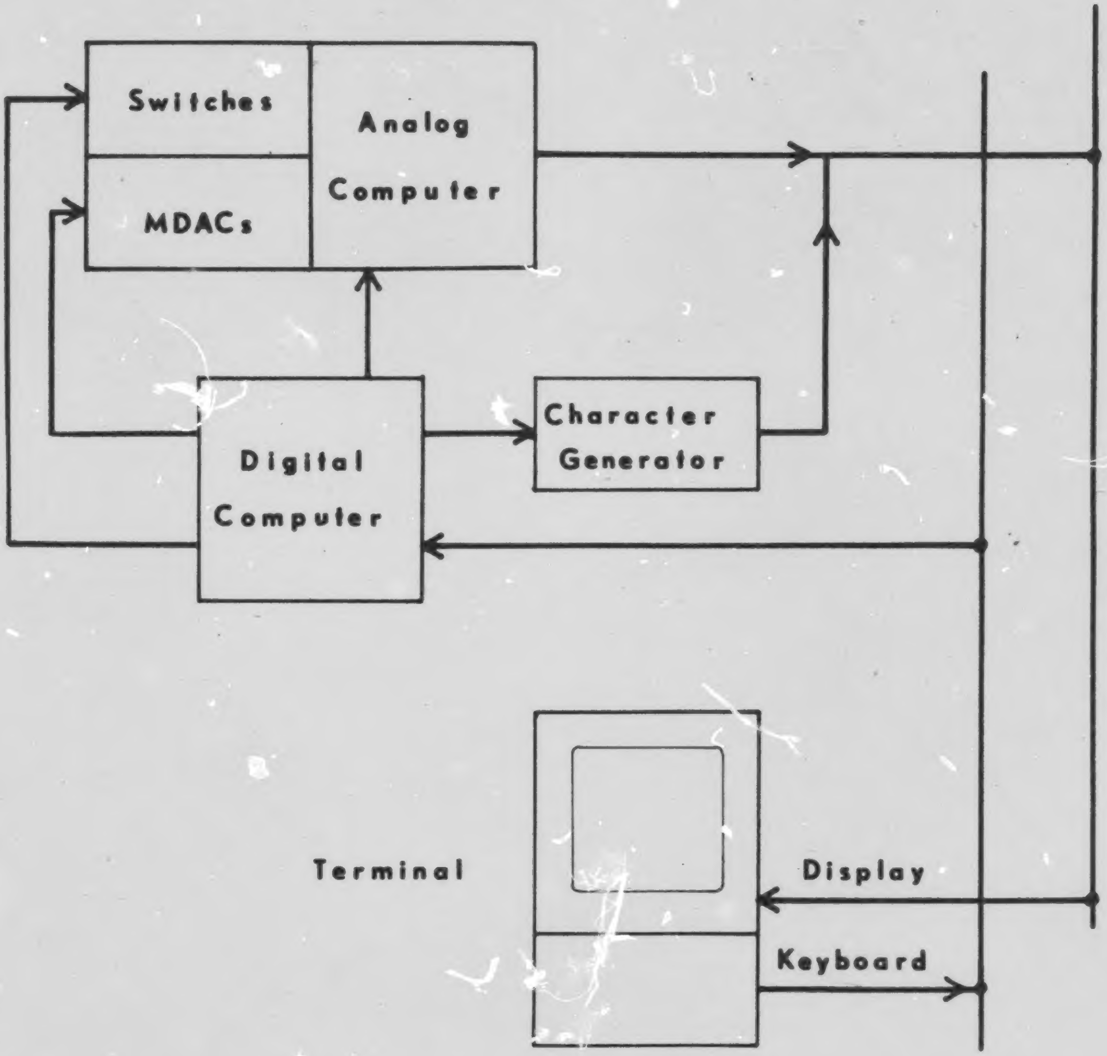

Switches
MDACs
Analog Computer
Digital Computer
Character Generator
Terminal
Display
Keyboard

Figure 1

AUTOPATCH I

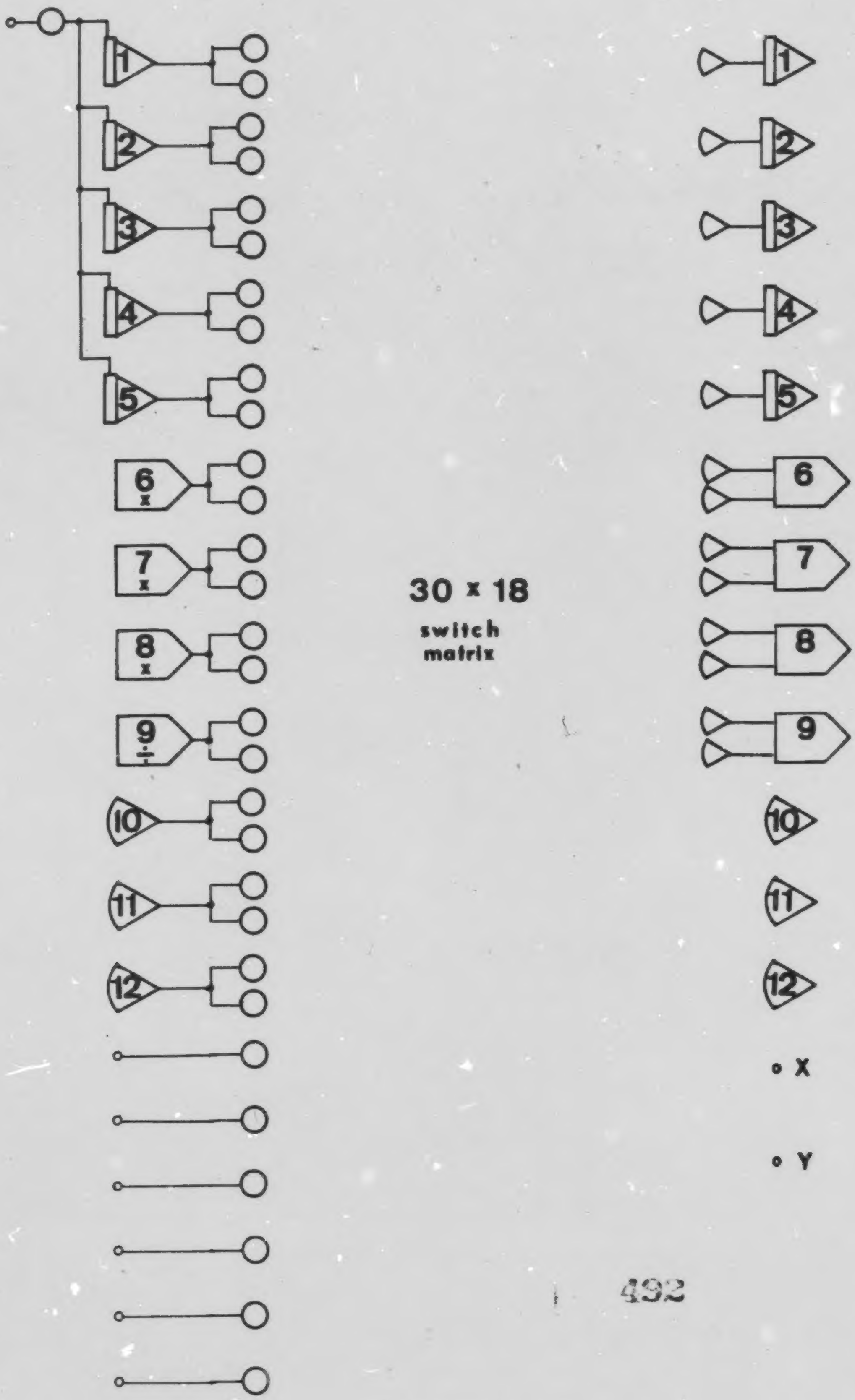

1
2
3
4
5
6
x
7
x
8
x
9
÷
10
11
12
30 x 18
switch
matrix
1
2
3
4
5
6
7
8
9
10
11
12
X
Y

Figure 2

ANALOG SWITCH

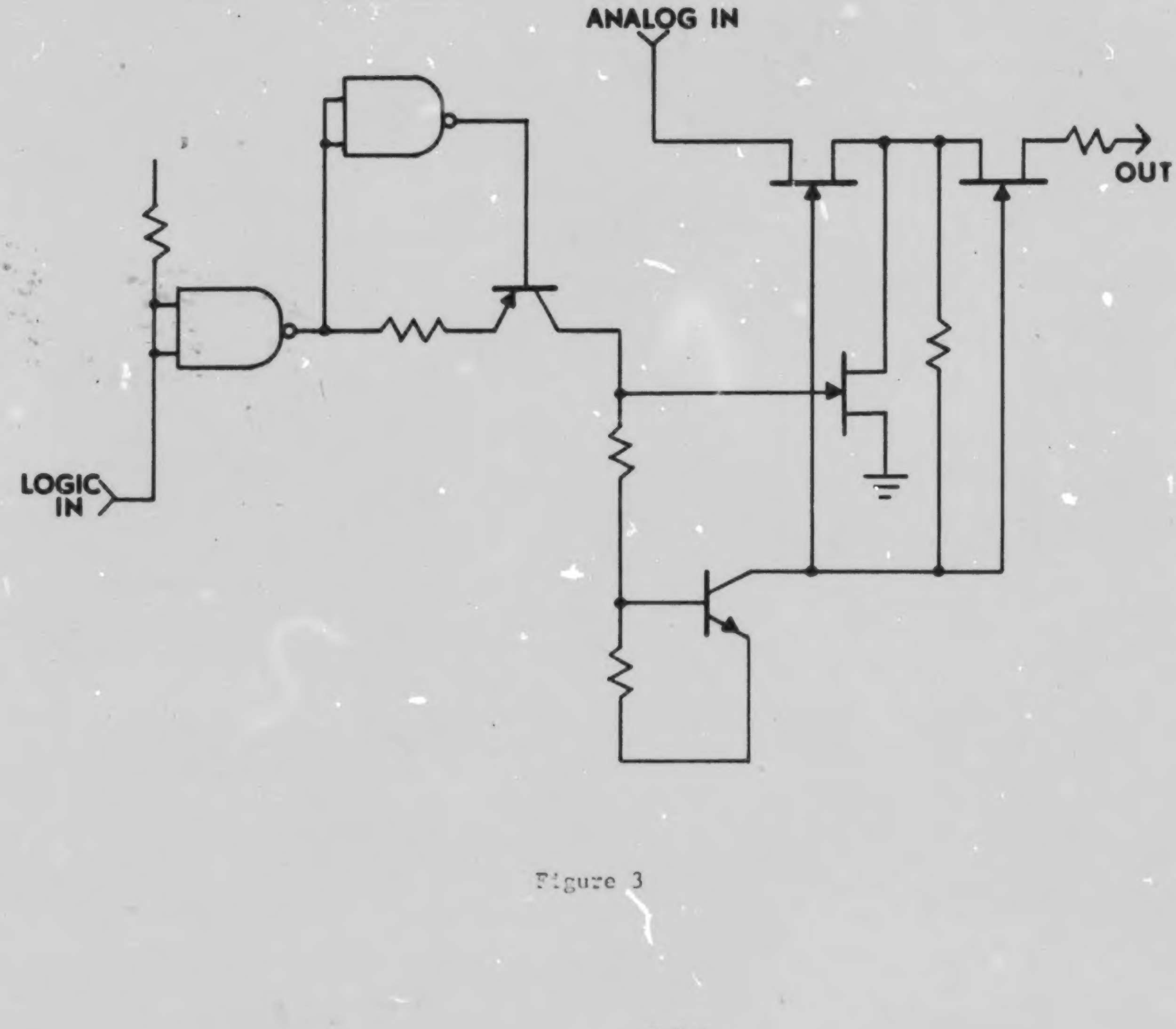
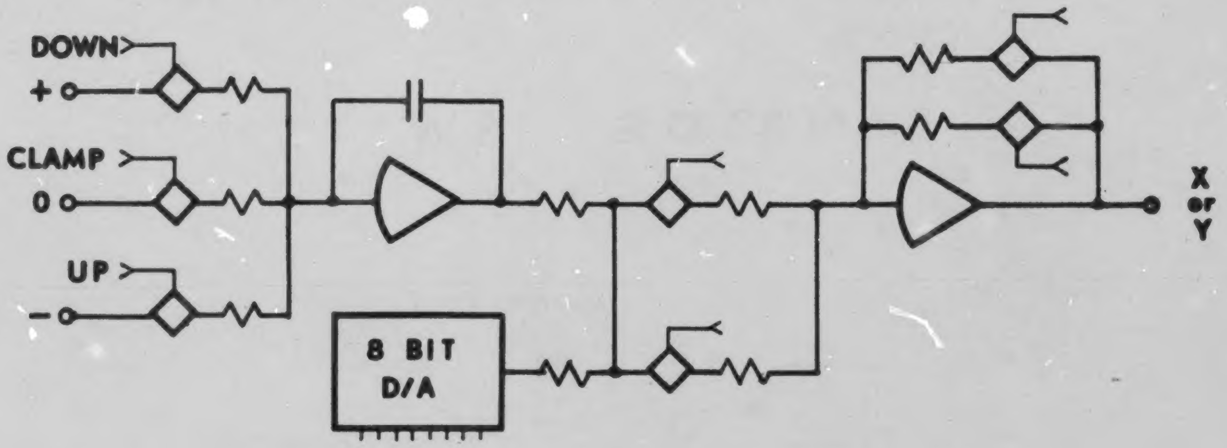

ANALOG IN
OUT
LOGIC IN

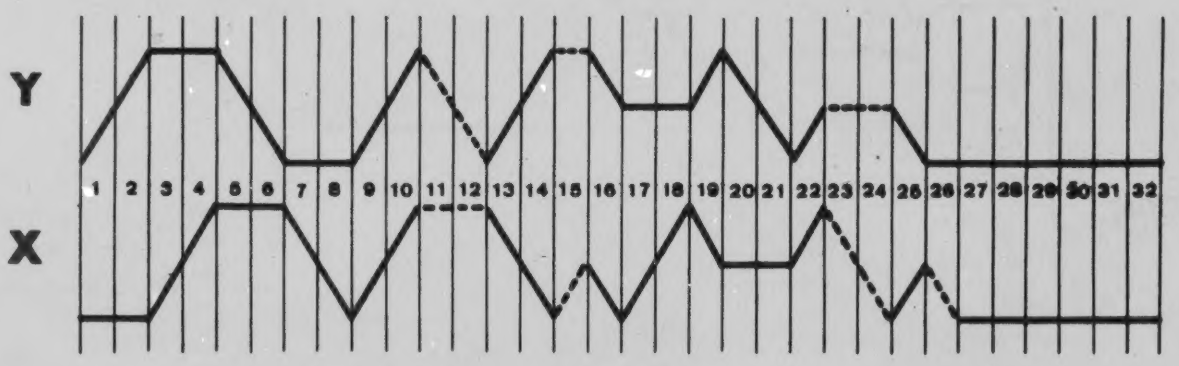
Figure 3

CHARACTER GENERATOR

a. Simplified Schematic of X/Y Generator


DOWN
+
CLAMP
0
UP
−
8 BIT
D/A
X
or
Y

b. X and Y versus Time Waveforms



c. X VERSUS Y PLOT

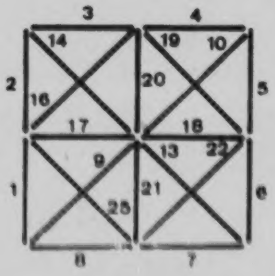

3
4
14
19
10
2
16
20
5
17
18
9
13
22
1
21
6
25
8
7

Figure 4

INTERFACE DETAILS

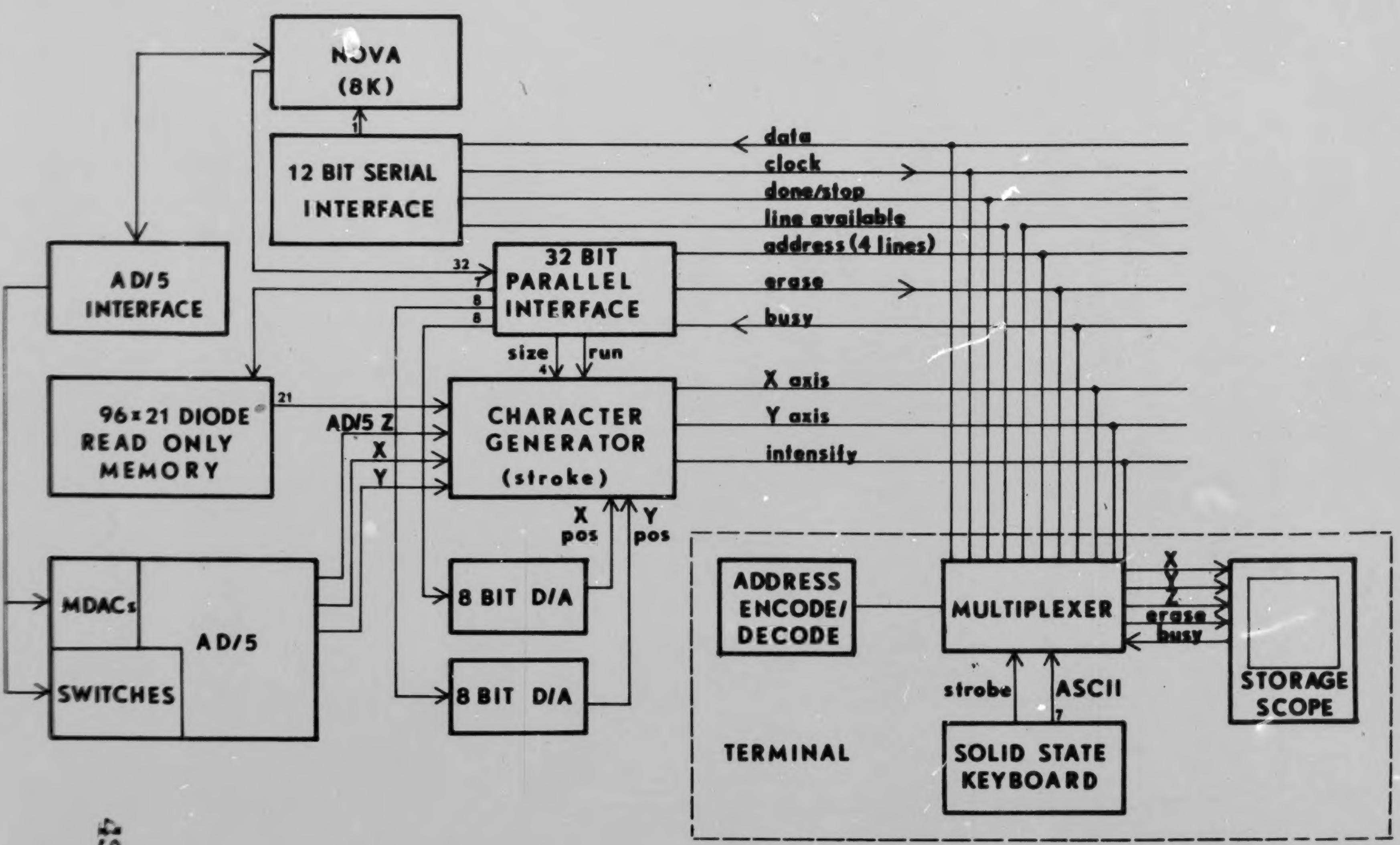

NOVA
(8K)
12 BIT SERIAL
INTERFACE
1
AD/5
INTERFACE
32
7
8
8
32 BIT
PARALLEL
INTERFACE
size
4
run
96x21 DIODE
READ ONLY
MEMORY
21
AD/5 Z
X
Y
CHARACTER
GENERATOR
(stroke)
X
pos
Y
pos
MDACs
SWITCHES
AD/5
8 BIT D/A
8 BIT D/A
data
clock
done/stop
line available
address (4 lines)
erase
busy
X axis
Y axis
intensify
ADDRESS
ENCODE/
DECODE
MULTIPLEXER
X
Y
Z
erase
busy
STORAGE
SCOPE
strobe
ASCII
7
SOLID STATE
KEYBOARD
TERMINAL

Figure 5

SAMPLER

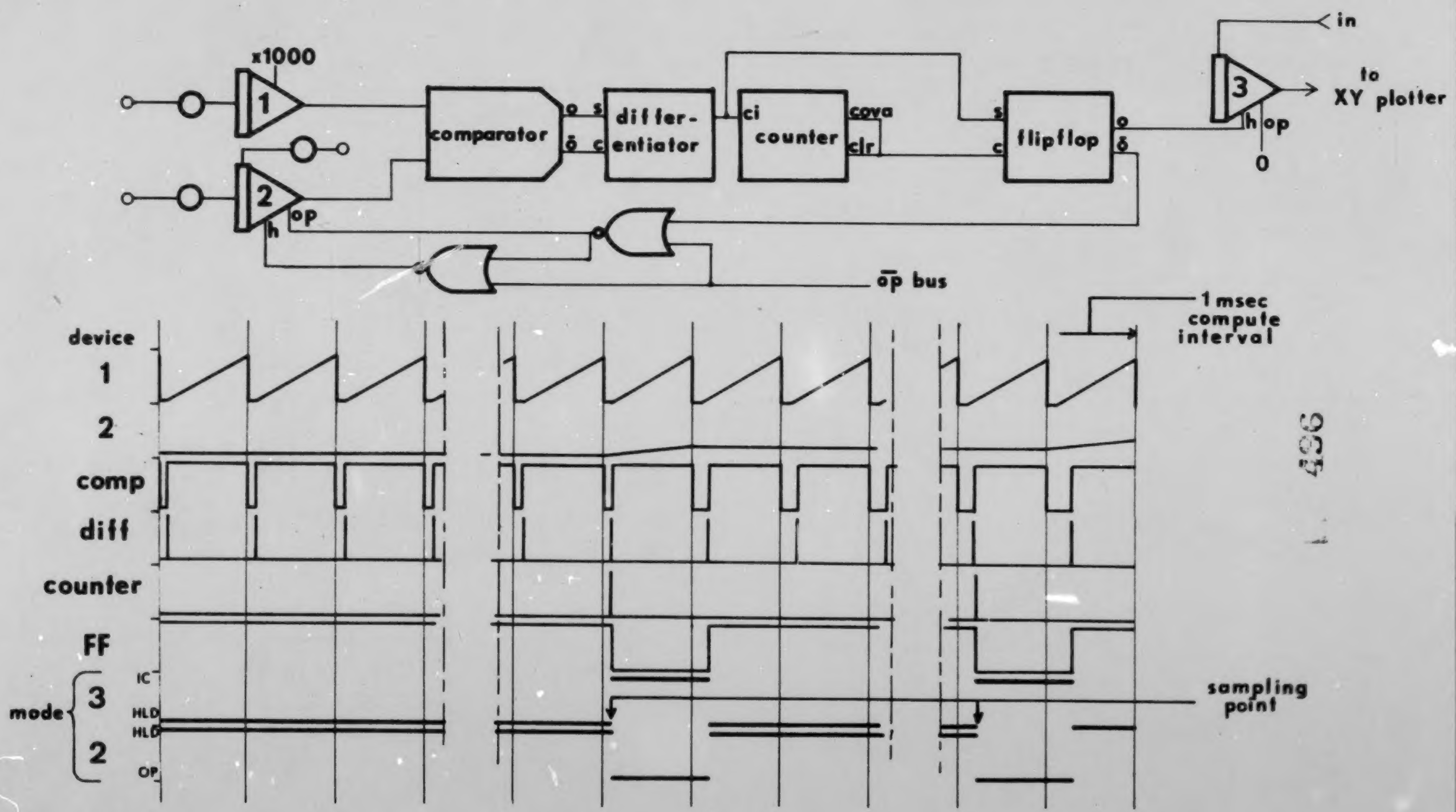

x1000
1
2
op
h
comparator
o
ō
s
c
differ-entiator
ci
counter
cova
clr
s
c
flipflop
o
ō
3
h
op
0
in
to XY plotter
ōp bus
1 msec compute interval
device
1
2
comp
diff
counter
FF
mode
3
2
IC
HLD
HLD
OP
sampling point

Figure 6

OVERVIEW OF SOFTWARE

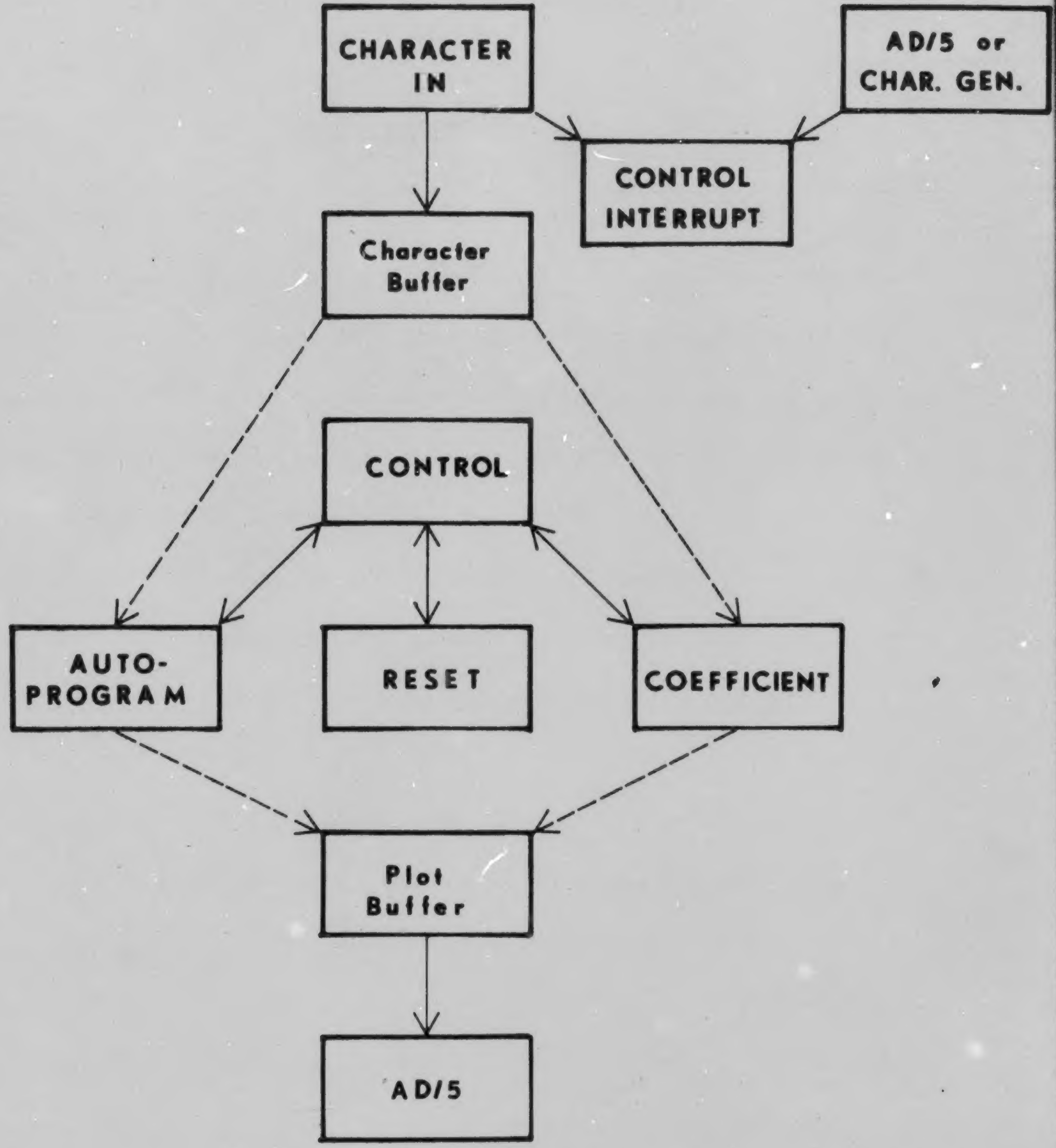

CHARACTER IN
AD/5 or CHAR. GEN.
CONTROL INTERRUPT
Character Buffer
CONTROL
AUTO-PROGRAM
RESET
COEFFICIENT
Plot Buffer
AD/5

Figure 7

Laboratory Simulation and CAI: A Rationale

M. T. Muller

Computer Based Instructional Systems, Inc.
Austin, Texas

Introduction

The use of computer modeling and simulation during the past twenty years evolved initially through military applications such as war gaming and human operations systems research efforts. Within the past ten years heavy emphasis on the use of simulation for training purposes in the space flight program has resulted in the development of sophisticated instrumentation that was designed for computer compatibility for real time applications. The resultant "spin off" of many of these simulation techniques and associated devices have been of a major benefit to commercial industry in adaptation to in-house training programs. Examples of the use of simulation are the lunar module flight simulator which allowed the astronaut to achieve a high degree of flight proficiency before the actual space flight.

Despite all of the progress achieved within the space program new innovative techniques in chemical education have been limited to traditional use of multi-media and more recently to computer based instruction. There is a bright note in this because many of the simulation techniques so successfully used in the space program can be usefully translated and applied to the field of laboratory instruction in chemistry. The purpose of this paper is to discuss in some detail the practical aspects of the use of computer based simulation (1) in undergraduate college chemistry laboratory instruction. The major objective of any teaching technique, be it traditional or innovative, is to make student learning a more meaningful and enriched experience in education.

Laboratory Simulation

It is commonly accepted that the role of laboratory work in undergraduate chemistry courses is to provide the student with opportunities (a) to use equipment and gain some degree of experience with laboratory techniques, and (b) for decision-making based on experimental data as well as for manipulation of these data. Thus, a technique may be taught in the laboratory and then belabored--apparently endlessly--to show the student how it may

---

1. A definition of computer simulation is quoted from "Computer Modeling and Simulation" by F. F. Martin, John Wiley and Sons, New York, 1968, page 5 as follows:

A computer simulation model is a logical mathematical representation of a concept, system or operation programmed for solution on a computer.

be used to obtain experimental results directed toward a variety of ends. For example, it is not uncommon to have students do acid-base titrations, equivalent weight determinations, oxidation-reduction titrations, complexometric titrations, analyses of various elements using titration, and methods analysis of biological systems. All of these experiments involve a basic titration technique, yet the student might spend numerous laboratory periods (in different courses) doing essentially the same thing, but for a slightly different reason each time. Similar situations exist in teaching many different experimental techniques.

Direct laboratory involvement is necessary to acquire manipulative experience; however, computer methods can be devised to provide experience in computational problem solving, decision-making and in the manipulation of experimental results. Furthermore, one of the characteristics of a university education is to develop the ability to interpret trends and concepts to bolster one's intuition during all of the above processes.

The logistics problems associated with laboratory instruction often impose an artificial constraint on the type of experiments that can be performed by the student. Laboratory instruction in many chemistry courses has not kept pace with the theoretical concepts introduced in the corresponding lecture portions of the same course for several reasons. First, the number of students involved has increased to the point where use of physical facilities must be severely restricted to accommodate everyone; schedules are crowded and logistics demand that experiments must be performed which require no more than three to four hours of laboratory time--a very artificial situation in the real world of chemistry. Second, funds are not generally available for the costly equipment required to perform experiments which would illustrate the more sophisticated concepts taught in the lecture portions of the course. Third, many experiments, although simple in principle, require several periods for their completion and cannot be performed for lack of time available to the student. The lack of time can often be traced to the factors described above and/or the idiosyncracies of the student's schedule.

Experiments Requiring Complex Equipment

Early in their careers many students are capable of handling the results of experiments which require apparatus too complex for their manipulative expertise. Indeed, to acquire a working knowledge of much of the apparatus used to obtain results which are commonly quoted in beginning science courses would require more time than is available; such experiments suffer the additional disadvantage that the equipment is usually available only for research purposes.

Emission spectroscopy is a very good example of an area from which modern general chemistry courses draw considerable material but a representative experiment is not usually included in the laboratory portion of the course. It is now possible to simulate a spectroscopic experiment using computer techniques (2) which can be performed in a relatively short time. With such a program available, early in his career, the student is required to underatand the principles on which a spectrometer works, make decisions concerning the data collected, as well as the manner in which the data will be treated, without becoming involved in the (often idiosyncratic) vagaries of the operation of a spectrometer. Presumably, the student will be exposed to the latter in a more advanced laboratory course when and if this experience is important to his professional development. Many classical experiments in the sciences can be placed in this category.

Time-Compression or -Expansion Experiments

Computer techniques are now being used to simulate experiments which take either unacceptably long or very short periods of time to perform. For example, there are many kinetics experi-

---

2. A course in spectroscopy has been programmed and demonstrated at the University of Texas at Austin on the IBM 1500 system by Lagowski and Castleberry in 1968.

ments that require relatively simple equipment (which is acceptable from a financial standpoint for large numbers of students); however, these experiments require periodic sampling for 10 to 24 hours which is unacceptable in a class. Experiments involving radioactive substances also fall into this category; in this instance, the cost of equipment as well as the time available to do the experiments can be simulated using computer techniques and performed on a more convenient time scale. Thus, computer simulation can be used to compress the time required for an experiment that might take several days, or expand time for an experiment that might take only a fraction of a second. A good example of this type of simulation is the Millikan Oil Drop Experiment. In either case, the experimental data could be handled by the undergraduate student even though the manipulation of the apparatus and/or the detailed planning of the experiment might be beyond his capabilities. An additional benefit arises from the ability of the student to process raw data rapidly when he attempts to fit such data to theoretical models.

Advantages of Computer Simulated Experiments

Computer techniques judicially used in conjunction with live experiments can lead to a more meaningful laboratory. Actual laboratory time can be gained if simulated experiments are used in place of live experiments which are performed repeatedly to gather data on different systems for decision making purposes; the time saved can be used to expose students to experiments which normally could not be performed. Using simulation, students can be exposed to relatively sophisticated experiments early in their training. In addition, some experiments can be extended into areas that would be impractical in a real laboratory, because the experiments are hazardous and/or the equipment too expensive.

Examplesof this include: a series of individual simulated experiments taken by students at the University of Texas served as a vehicle for demonstrating the unity of experimental work directed toward a given goal. In the chemistry laboratory, simulated experiments involving qualitative analysis, certain qualitative techniques, molecular weight determinations, and spectroscopy were organized for the purpose of identifying and characterizing unknown substances. An attempt to do this in actual practice would undoubtedly involve more time, equipment, and space than would be available in a normal laboratory course. However, the equivalent in computer simulated experiments was accomplished in several afternoons. Indeed, it is predicted that in the evolution of these techniques future laboratory courses will be developed which incorporate a subtle blending of real laboratory experiments and computer simulation to the point where the student will at times feel the two are inseperable!

Finally, it can be expected that considerable savings in terms of space, equipment, and expendable supplies would accrue from laboratory simulation techniques. Computer simulation can be used to advantage to alleviate many problems associated with laboratory instruction of relatively sophisticated material to large numbers of students.

Computer Control of Experiments

The nature of experimental work in modern chemistry is being greatly influenced by the development of computer systems that can intervene directly in real time with the course of an experiment. It is now possible through computers to collect several kinds of data virtually simultaneously, process the data in a standard way, make a judgment (either by the experimenter or the computer) of the results, and intervene in the experiment while it is still in progress. Such applications now are common in industry, but the undergraduate junior or senior in the average advanced laboratory course knows virtually nothing of such computerized data collection techniques. The development of computer controlled (or monitored) experiments for use in advanced undergraduate laboratories is but one of the newer capabilities urgently needed.

Several years of experience obtained in the application of computer based techniques in general chemistry and in organic chemistry strongly suggest that these methods could be successfully applied to every level of instruction in technical and vocational courses. A description of the individual modules developed for introductory chemistry courses by Computer-Based Instructional Systems, Incorporated, is shown in Appendix A. A discussion of a typical example of the use of simulation techniques will now follow.

PHOMA (See Appendix B)

This module is a simulated experiment in the use of spectrophotometers for quantitative analysis. The procedure is based upon that described in ASTM, Part 32, pp 78-80, May, 1965. Essentially the experiment requires that the student demonstrate he is familiar with the outlined procedure and then collect and analyze data as if he were in a real laboratory environment. The students' results are then examined for accuracy and recorded through the use of a sister program, PERMA. It will be necessary to assign each student an "unknown" number. The "actual" value of the unknown sample will be based upon the number assigned as follows:

Any number less than:	%Mn
1000	0.017
2000	0.41
3000	0.35
4000	0.49
5000	0.62
6000	0.75
7000	1.25
8000	1.45
9000	0.82

A random experimental error is generated to avoid total duplication of results.

A typical student interaction follows. (Student input is capitalized and underlined for clarity.)

Your friendly CBIS chemistry tutor bids you welcome.

Please enter your student number ? 2

Thank you, Mark

Which of the programs listed in your index would you like to take today? Type in the name of your choice? PHOMA
The Spectrophotometric Determination of Mn in Steel

Have you read the experimental procedure? YES
Good...what is the maximum percent of Mn that is suitable for this experiment? 5

O.K., swell...at what wavelength in millimicrons will measurements be taken using the spectrophotometer? 545

Great...we are on the same wavelength...what is the substance that imparts color to the solution that is analyzed spectrophotometrically? PERMANGANATE ION

Mighty fine...I really believe that you are ready to perform this experiment...

Now please enter your assigned experiment number? 1500

What weight (in grams) of 99.9 percent pure Mn do you wish to dissolve in hot HNO_3 in preparation of the standard? 2.0
---you are out of the optimum range for preparing the standard from pure Mn. Please examine the experimental details and answer again...
What weight (in grams) of 99.9 percent pure Mn do you wish to dissolve in hot HNO_3 in preparation of the standard? .5
Fine...the sample has been dissolved in a minimum of hot HNO_3 and allowed to cool to room temperature. To what volume (in ml) do you wish to dilute this solution? 100

The concentration of solution 'A' is now 5 mg Mn/1.00 ml.

What volume (in ml) of solution 'A' do you wish to dilute? 1
To what volume (in ml) do you wish this aliquot diluted? 150

*****Your standard solution contains 0.0333 mg Mn/1.00 ml.

You might want to check this calculation at the conclusion of the experiment to verify my results

Very good...now I will take whatever volume of this standard you would like, add KIO_3, dilute to 50 ml, heat for 30 minutes to develop the color and allow to cool to room temperature.

Now please enter separately 5 volumes of the standard solution to be so treated...

```
Calibration solution volumes:
        #1, ? 5
        #2, ? 10
        #3, ? 15
        #4, ? 25
        #5, ? 30
```

Thank you...the solutions have been treated to develop the color and analyzed spectrophotometrically. The following absorbances have been obtained...

```
   Solution    --- Absorbance
***** # 1      ---0.21
***** # 2      ---0.33
***** # 3      ---0.47
***** # 4      ---0.71
***** # 5      ---0.87
```

*****The cell correction for absorbance is found to be 0.03

O.K., let us now gather data about the unknown sample...
*****weight of steel sample   beaker = 3.0321 grams
*****weight of beaker = 2.5916 grams

I have now prepared the unknown sample for analysis and diluted it to 100.00 ml as is described in the experimental procedure.
.....what volume (in ml) of this solution do you wish to use as an aliquot.....

*****sample aliquot? 10

To what volume (in ml) do you want to dilute this aliquot...

*****sample dilution? 50

```
The diluted unknown sample solution has been analyzed with a
spectrophotometer...

***** the absorbance of the sample solution is 0.29
***** the cell correction for absorbance was found to be 0.01

Determine the percent of Mn in your sample.  When this is completed,
login with program 'PERMA'.  I thank ye kindly.
```

The students' experimental results are then checked by the following program.

PERMA

In the program, the student enters:

1. His assigned experiment number
2. Plot or ratio determination

3a. Mg of Mn from the curve, or,

3b. Corrected absorbance of the unknown sample, and if 3b.,

3c. Ratio: mg Mn::absorbancy

4. Sample aliquot
5. Sample dilution
6. Sample weight

A check of the students' calculated percent and actual percent is then performed and the results output to the student. This includes a theoretical "grade" based upon:

Difference: Cal'c & True %	Grade
< 0.021	A
< 0.051	B
< 0.11	C
< 0.15	D
> 0.15	F

A sample student interaction follows.

```
Your friendly CBIS chemistry tutor bids you welcome.

Please enter your student number? 2
Thank you, Mark

Which of the programs listed in your index would you like to take
today?  Type in the name of your choice? PERMA
```

```
Welcome back...
Enter your assigned experiment number? 1500

Did you determine the percent of Mn by:
        1. plotting absorbance vs mg Mn/50 ml
        2. ratio of mg of Mn to absorbancy
Please answer by number.
?2
Complete the following:

***** Corrected absorbance of sample = ? .28
***** Ratio: mg of Mn::absorbance = ? 1.20
***** Sample aliquot (ml) = ? 10
***** Sample dilution (ml) = ? 50
***** Sample weight (grams) = ? .4405

Well now, I calculate your percentage as 0.381 and the actual
percentage of the unknown is 0.41. Not bad, not bad at all, old
cold hands, for if I'm scoring your results, your grade = B

Thanks for being here, partner...
```

It should be emphasized that although this paper may appear to be cast in a static context, the state of the art is rapidly changing because innovations in hardware and software are characteristic of computer technology. Experience in using computer-based instructional modules is being gained continually with larger and larger groups. The amplification and daily acceptance of these innovations on a broad spectrum of subjects is a goal that is within sight and will become a reality in time. How rapidly this will be accomplished, one cannot say for change in education is a slow evolutionary process that is tempered with caution and conservatism to an extreme degree. The final decision for acceptance will rest with the educator be it good, bad or indifferent!

APPENDIX A

Module	Area	Mode
1. MET1	The Metric System	Drill
2. MET2	The Metric System	Applied Drill
3. TEMCO	Temperature Conversions	Drill
4. DECAL	Density Calculations	Applied Drill
5. PETEL	Percentage Calculations	Applied Drill
6. GAS1	The Gas Laws	Simulation
7. GAS2	The Gas Laws	Applied Drill
8. EQBAL	Balancing Equations	Drill
9. FORM1	Formula Writing	Drill
10. NOBIN	Inorganic Nomenclature	Applied Drill
11. NOTER	Inorganic Nomenclature	Applied Drill
12. NOANE	Organic Nomenclature	Tutorial Drill
13. QUANE	Organic Nomenclature	Examination
14. NOGEN	Organic Nomenclature	Applied Drill
15. COPRO	Colligative Properties	Simulation
16. PHOMA	Instrumental Analysis	Simulation
17. PERMA	Data Check of PHOMA	--
18. BLEAP	Instrumental Analysis	Simulation
19. PERCH	Data Check of BLEAP	--
20. GRAVE	Quantitative Analysis	Applied Drill

APPENDIX C

A Partial List of Institutions Utilizing Laboratory Simulation Techniques

Institution	Developer(s)
University of California at Santa Barbara	Ewig, Gerig, and Harris
Smith College	Fleck
The University of Texas at Austin	Castleberry, Culp, Lagowski
Carleton College	Child, Finholt, Ramette
University of Oregon	Klopfenstein, Wilkins
University of Pittsburg	Johnson
Washington State University	Willett, Breneman, Kissler
College of St. Elizabeth	Sister Callavan
Xavier University	Corrington, Marshal
N. Carolina Educational Computing Service	Denk
Fairmont State College	Swiger
Florida State University	Mellon, Dence, Graham
University of Illinois	Smith

USES OF COMPUTERS IN COMPENSATORY COLLEGE-LEVEL SCIENCE EDUCATION

by

Ralph M. Deal
Chemistry Department
Kalamazoo College
Kalamazoo, Michigan

INTRODUCTION: COMPENSATORY EDUCATION AS A CURRENT HIGH PRIORITY NEED IN HIGHER EDUCATION.

Most current interactive uses of computers in programmed learning are too expensive to use on a scale larger than research or developmental. One area where the use of a computer terminal by a student or a small group of students to develop specific concepts and skills is economically feasible is compensatory education. I am referring specifically to most institutions of higher learning in the United States bringing black students and spanish-speaking students into their freshman class although these students do not meet the usual entrance criteria.

For the most part, these institutions have felt that they have done their part to combat institutional racism by confronting their usual white students with the presence of black peers. Unfortunately, these new students are not academic peers. They come from educational backgrounds in which proper preparation for survival in the university or college is not possible. These students then do poorly in their classes, reinforcing the biases of white students and black students. "Liberal" professors who give underprepared students higher grades than they deserve are "compensating" in a way damaging to those students. Increasingly, colleges and universities have come to realize the failure of their current programs to meet the educational needs of underprepared students and are preparing to either (1) give up accepting students who do not meet the usual academic requirements (especially entrance tests such as SAT) or to (2) commit a significantly larger share of their institution's resources to preparing the new, illequipped student to compete favorably in regular classes. Since this is a new task, no one is sure how best to provide the required "compensatory education". Certainly one method worth exploring extensively is the use of highly individualized learning programs with some way of pushing each student to use all his resources to solve problems in a wide variety of situations and, at the same time, supporting and encouraging the student with a high success rate. This is probably best done with a combination of proctoring by sensitive, knowledgeable, self-confident peers and a series of programmed units ideally administered through a computer. The course structure developed by Fred Keller and applied to many disciplines (including physics at the Education Research Center at MIT and at Portland State University) seems appropriate.

ADVANTAGES OF A COMPUTER IN COMPENSATORY EDUCATION.

What is required is a way to achieve the very difficult task of taking a student back over skills he has been exposed to and feels he knows, show him where he needs to rework his knowledge and skills and let him know when he has mastered them, all in a few months during which his suburb-trained peers are progressing through courses he would find very difficult to pass.

The computer has several advantages over programmed textual materials here: (1) The student cannot cheat (himself), (2) the time taken for each frame can be used to push the student to work faster, (3) detriled records of student performance can be used to improve the program through continuous modifications. From my experience, well-written programmed materials are frequently not used effectively by students who are suspicious of the nit-picking habits of academia and accept too easily their own answer as equivalent to the correct response given in the text. Of course, the computer is, at this stage in the development of CAI, not going to recognise an original valid response the first time it is encountered, any more than a programmed text. However, with a sufficiently flexible learning system and a dedicated programming staff (programming in both computer and educational sense), a personal response can be made to the student giving original insightful responses and the computer programs can avoid the rejection of subsequent similar responses.

HISTORY OF A COMPUTER AUGMENTED COMPENSATORY COURSE.

In the spring and summer quarters of 1969-1970 at the Chicago Circle Campus of the University of Illinois, I ran, with Barry Clemson of the Educational Assistance Program of the Circle Campus, an experimental chemistry course entitled, "Problem Solving in Chemistry." Based on the analysis by Robert I. Walter of the Chemistry Department of the University of Illinois at Chicago Circle of pretests of chemical knowledge and mathematical skills and subsequent performance in introductory chemistry courses, we decided to emphasize math skills rather than chemical concepts in our compensatory course. Throughout the course, the students used computer programs in groups and individually on a 2741 terminal to a 360/50. All of the programs were written in CPS, a conversational subset of PL/I, the only conversational terminal language available in the Chicago Circle terminal network (other than a form of BASIC). Many of the programs used were developed with individual students in tutorial situations in the fall and winder of 1969-70. The interpretive nature of CPS was taken advantage of in the operation of several programs as well as in the construction of all of the programs. The programs are described briefly in the appendix.

In the fall of 1970, I ran a similar course at Kalamazoo College, using a teletype terminal to a PDP-10 at Western Michigan University, using principally BASIC. I was not able to develop as wide a range of programs at Kalamazoo College as I had used at the Circle Campus but I introduced a learning sequence in BASIC itself which seems promising in terms of student attitudes in this course.

PROCTORING IN COMPENSATORY EDUCATION.

Since, in my experience, students in compensatory education programs need an extraordinary amount of support from a sympathetic proctor, no computer learning system should be expected to work by itself with a single student. The Keller system provides one proctor for 10 students. No more than 10 students per proctor during each session seems to be important.

Students work well in groups of roughly equivalent ability and provide much of the necessary support for each other. This mode of operation is most effective when a proctor can frequently act as a group observer and make occasional comments on the dynamics of the group. Each member of the group is charged with the responsibility of being sure that the other members of the group are fully participating. This behavior can be reinforced by requiring all the individual members of the group to pass an individual mastery test before the group can go on to the next program or topic.

LANGUAGE CONSIDERATIONS.

Whether CAI or other learning materials and media are used in such a course, they will require significant modification to serve underprepared students well. Since new materials must be developed from day to day, a flexible preprocessor of a higher level language is very desirable as is an interpretive language. A user's group of educators using computers in compensatory education should be formed to share the experience and programs of such developmental programs. Eventually we should have a shared set of programs from which a program can be selected to fit the particular needs of a particular student. Of course some standardized language such as PLANIT would be most useful since it would allow a program developed at one institution to be used by another. PLANIT, COURSEWRITER and other CAI language save programming time but limit the form of the student-computer interaction. I have found BASIC and AID on the PDP-10 to be adequate for simple learning programs at Kalamazoo College. The advantage of using conventional languages is their being available on small computers and requiring only minimal core on a large computer. Therefore, they can be used by small institutions who have their own small time-sharing system or are purchasing time on a time/sharing network which will not maintain a large CAI language such as PLANIT or COURSEWRITER on the public library.

In the winter quarter of 1971 I worked with a vastly improved version of PIL (Pittsburgh Interpretive Language) at the University of Michigan. James Conklin at the Center for Research on Learning & Teaching in Ann Arbor, Michigan, has expanded PIL to include useful commands for string-manipulation and the capability of calling FORTRAN or assembly compiled subroutines. I find that this combination of a powerful interpreter dealing with students directly with much of the work being done in compiled programs to be flexible and efficient and to allow rapid program modifications.

CONCLUSION.

If universities and colleges are serious about compensatory education, they must make a considerable investment in special programs designed to equip underprepared students to compete favorably in regular courses. This investment can be in personnel and/or hardware. Hardware alone is not enough, but computers can be an effective part of such an effort.

APPENDIX: Brief description of computer programs written for use in compensatory courses at Chicago Circle Campus of the University of Illinois and at Kalamazoo College.

PROGRAM	LANGUAGE	BRIEF DESCRIPTION
CAI/CPS	CPS	A flexible CAI processor with most of the features of COURSEWRITER and quite a few more features. There are two basic modes of operation - author and student. The author mode is designed for teachers with no prior computer experience. The main advantage of this particular system is the flexibility in the kinds of patterns it can recognize in the student responses. The structure is capable of some syntactical analysis. The student interaction is unfortunately much too slow to be an effective form of CAI.
Line Length	CPS	A very simple program that prints out a pair of *'s spaced at random intervals across the page. The student is asked to estimate distance between the *'s in several different length units. The student is given the ratio of his answer to the correct length in percentage. When the student uses this information he is of course doing drill and practice in percent as well as line length estimation.
Slide Rule	CPS	A generative drill and practice routine designed to improve the speed and accuracy of the student's use of a slide rule. Several parameters may be varied by the student or proctor to increase the difficulty of the task. The computer counter-responds with "decimal point error", "decimal point OK", "close", "very close", "GREAT!", depending on the student response. The time taken from the presentation of the problem to the correct student response is printed out in an attempt to encourage faster slide rule calculations.

PROGRAM	LANGUAGE	BRIEF DESCRIPTION
Test & Drill	CPS	The student selects the kind of problem (e.g., first degree equations, negative numbers). A pretest of generated problems, two of each of 3 to 5 types in each problem set, is given and scored. Problem types missed are then presented until 5 are correctly answered in a row. Then a posttest is used to measure any improvement in student performance. A printout of the interaction is provided for both student and instructor.
Symbol Manipulation	CPS AID	Algebraic equations are presented with the request to rearrange the equation to solve for one particular variable. A variety of symbols may be used. If the student fails to provide a valid rearrangement, a correct rearrangement is presented.
Work problems	CPS	This is an attempt to produce an economical CAI system within a very limited context of word problems requiring numerical answers with units. The problems themselves are in a separate booklet. The computer only accepts and evaluates student answers. The format for the units is quite limited.
Dragster Simulation	CPS	A digital implementation of a hybrid computer program written by EAI. Three coupled second-order differential equations are involved. Program is both slow and inefficient - the hybrid obviously does this job better.
Dragster - Simple Model	CPS BASIC	A very simple mathematical model of the finish time of quarter mile drag race as a function of 7 variable parameters (such as wheel base). Possible outcomes are SLIP, FLIP, FINISHED IN 13.7 SECONDS AT 146 MILES PER HOUR. This program was used to explore functions of one and two variables.

ARMCHAIR UNKNOWNS

Clifford G. Venier and Manfred G. Reinecke
Department of Chemistry
Texas Christian University
Fort Worth, Texas 76129

We would like to report our experience with a computer simulated organic qualitative analysis course as a teaching aid in undergraduate organic chemistry over the past two years. Previously, Smith (1) has outlined a dialogue format for this type of course between student and the University of Illinois PLATO system, but not all of us have PLATO as a helper. Gasser and Emmons(2) described a system for using the computer to test results from the laboratory which is similar to computer simulated unknowns, but emphasizes a different aspect of computer use, namely data sorting. Our own system derives from that described by Swets and Feurzeig(3) many years ago, which uses a computer to teach medical diagnosis, a subject very similar in nature to organic qualitative analysis. The underlying idea of any such computer exercises is that a real patient need not die in order to allow a novice to learn through correction of his mistaken inferences. In an organic qualitative analysis simulation, the student is not burdened with shortcomings in his conception of the logical framework of analysis and, at the same time, misinformation gleaned from his incompletely developed techniques. Hopefully, in his first semester laboratory, the student has gained a modicum of confidence in his basic experimental skill. Our goal is to simulate the lab experience which the student will encounter as closely as one can on CAI CRT's with a Coursewriter program on an IBM 1500 computer.

Being inexperienced in the use and programming of computer-assisted instruction, we have also attempted to keep things as simple as possible, consistent with our goals. For this reason, we decided to tie our course to a single text, R. L. Shriner, R. C. Fuson, and D. Y. Curtin, "The Systematic Identification of Organic Compounds," Fifth Edition, Wiley, New York, 1964(SFC). We chose this text as the most popular and widely used of its genre.

Coded responses are far more easily programmed than the keyword recognition dialogue described by Smith(1), yet allow immediately a far more extensive set of experiments and unknowns than the program described by Gasser and Emmons(2). Shriner, Fuson, and Curtin, as a text, is easily coded. We have done so as follows:

1. Compounds(unknowns) are given tags, mm-nnn-x, where mm is the table number of the table in which the compound appears in the Tables of Derivatives on pages 307-382 of SFC, nnn is the boiling point, melting point, or decomposition point of the compound, whichever is listed first in Table mm, and x is an index in case of multiple listings under mm-nnn, simply denoting that the compound is the xth one with mm-nnn. For example, 2,4-dimethylbenzenesulfonyl chloride is the second compound with melting point 34°C. listed in Table 45. Its code is 45-34-2. Methyl benzyl ketone is the first, and only, compound in Table 38 with a melting point of 27°C. Its code is 38-27-1. Its boiling point of 216°C. is listed after the melting point and may not be used in the code.

2. Chemical tests are divided into three categories.
   a. Preliminary tests are individually coded.
      ig - ignition test
      mp - melting point
      bp - boiling point
      sg - specific gravity
      nd - refractive index
      ad - optical rotation
      mw - molecular weight(Rast)
      sf - sodium fusion
      sl - solubility
   b. All classification tests are coded c-yy where yy is the number of the experiment outlined on pages 112 to 174 of SFC. For example, the Tollen's test is c-29, the chromic anhydride test for primary and secondary alcohols is c-8.

c. All derivative tests are d-zz, where zz is the procedure number from pages 229 to 304 of SFC. For example, a 2,4-DNP is d-18, and an amide of a carboxylic acid is d-2.

The program itself is presented to the students entirely in the first two weeks of the second semester. This is imperative since they must have confidence in their logical approach to their laboratory unknowns as soon as possible. This procedure has the added advantage that other CRT users usually don't need computer time during the first two weeks of their courses, allowing one to have twenty to thirty hours of time each of the first two weeks. Each student is first instructed and drilled in the use of the keyboard and light pen, then taught the codes and how to use them. As a preface to his computer unknowns, he is given a trial problem and assured that no score is kept on his progress through it. In fact, he is encouraged to play with the system a little to see what it will and will not do. At this point, the student will either flounder or start to use his book. He must read and understand the introductory material in the book before he can make very much progress on this trial problem. After completion of the trial problem, he is assigned three unknowns to identify. So far, we have given each student a carbonyl compound, another neutral compound, and an acid or a base.

The CRT's are made available, usually in the evening or on weekends, on a first-come, first-served basis. The student may leave the computer at any time and return later without penalty. We encourage the student to sign off if he takes more than a few minutes between tests. Typically, a student will get five minutes' worth of data from the computer, then leave, returning later after interpreting the facts with the aid of the lab text, do one or two additional tests and repeat the process until he has identified the unknown. A typical student-computer dialogue is as follows.

CRT: Do you have the handout sheet on which the codes for the experiments are listed? If not, get one from the assistant before you continue. You will also need the Shriner, Fuson, and Curtin textbook.

CRT: When you have a copy of the mimeographed sheet with the codes of the experiments on it, and the textbook, then press the spacebar to continue.

CRT:* The unknown is a solid. Which test or experiment do you wish to make now? Type in one of the codes below.

Student: ig (ignition test)**

CRT: burns with a smoky flame.

CRT: The unknown, etc.

Student: mp (melting point)

CRT: 31-34°C.

CRT: The unknown, etc.

Student: sf (sodium fusion)

CRT: All tests negative.

CRT: The unknown, etc.

Student: sl (solubility)

CRT: H_2O, no; HCl, no; NaOH, no; H_2SO_4, yes.

CRT: The unknown, etc.

Student: off(signed off the CRT temporarily)

---

* This statement is repeated after each test result is displayed. It will be abbreviated, "The unknown,etc."

** For the reader's convenience, the interpretations of the codes are included in parentheses.

The CRT displays an INDEX when the student returns and the student uses the light pen to return to his unknown.

CRT: The unknown, etc.

Student: c-9 (2,4-dinitrophenylhydrazine)

CRT: deep red-purple ppt

CRT: The unknown, etc.

Student: c-29 (Tollen's test)

CRT: silver mirror deposited

CRT: The unknown, etc.

Student: off

The CRT displays an INDEX when the student returns and the student uses the light pen to return to his unknown.

CRT: The unknown, etc.

Student: d-16 (semicarbazone)

CRT: white solid, mp 225-227°C.

CRT: The unknown, etc.

Student: d-18 (2,4-DNP)

CRT: reddish-violet solid, mp greater than 250°C.

CRT: The unknown, etc.

Student: bp (boiling point)

CRT: approximately 250°C.

CRT: The unknown, etc.

Student: d-49 (oxime)

CRT: white solid, mp 108-109°C.

CRT: The unknown, etc.

Student: 22-37-3 (piperonal)

CRT: Excellent, You have arrived at the correct answer. Next you will be shown the INDEX of unknowns available. Use your light pen to choose one.

Although no formal evaluation of this program has been attempted, it appears that the students work much more efficiently on their laboratory unknowns after exposure to our CAI organic qualitative analysis scheme. For example, while many students were not able to finish five unknowns per semester prior to CAI, they are now able to solve six unknowns with no loss in accuracy of identification. We are pursuing modifications designed to make the simulation even more realistic.

Acknowledgements:We wish to thank the staffs of the Instructional Systems Institute and the Computer Center of Texas Christian University, especially Mrs. Theresa Holt and Mr. Jeremy Main, for valuable assistance.

REFERENCES

(1) S.G. Smith, _J_. _Chem_. _Ed_., 47, 608(1970).

(2) W. Gasser and J.L. Emmons, _J_. _Chem_. _Ed_., _47_, 137(1970).

(3) J.A. Swets and W. Feurzeig, _Science_, 150, 572 (1965).

Chemistry and CAI at the University of Texas at Austin: A Coordinated Approach

G. H. Culp
The University of Texas at Austin

Introduction

Utilization of computer techniques in undergraduate chemical education has been an on-going process at the University of Texas at Austin since 1965. The first carefully controlled educational experiments involving the use of CAI as a supplement to general and undergraduate organic chemistry were conducted there during 1968-70.[1-7] The results of these studies indicated a consistent, highly positive effect on student performance and attitude when computer-based techniques were applied. These studies have also provided an excellent foundation upon which to base current efforts toward the actual implementation of supplemental CAI programs in undergraduate chemical education.

The testing and evaluation of the original programs was conducted through the cooperation of the University of Texas Computer-Assisted Instruction Laboratory and utilized the IBM 1500/1800 computer system. As these studies progressed, several limitations with the available facilities became apparent. First, by definition, the CAI laboratory is dedicated to research. Often several concurrent projects in various educational disciplines are in development within the laboratory, thus severe limitations are placed upon the number of student participants in a given project. Likewise, the ease of access to the programs is confined by the large number of projects and the limited facilities. Secondly, the Coursewriter II language was found to be inappropriate for several types of potential CAI applications that require direct computational capability. And, in addition, the programs could not be teleprocessed to remote terminals due to hardware video data transmission limitations.

It was evident that both a more versatile language and a more flexible system would be necessary to sufficiently realize the CAI potential applied to undergraduate chemistry courses.

The Development of CLIC

As an outgrowth of the documented effectiveness of CAI and the need of a language that would provide the capability for CAI utilization on a broad basis within the University of Texas system, computation center personnel developed the language, CLIC (Conversational Language for Instruction and Computing).

The CLIC language bears a number of similarities to the well-known Coursewriter-type languages in its use of a very simple statement format and question-answer logic. The language is actually an extension of the PICLS language developed at Purdue University. The basic CLIC language is very easy to learn and can be readily mastered with only a few hours instruction. Thus the beginning user quickly reaches the point where he is writing simple instructional programs.

For the experienced user, however, CLIC differs substantially from both PICLS and COURSEWRITER. CLIC is designed as an extension of the general purpose programming language, FORTRAN. Thus the CLIC user has access to all of FORTRAN, including the full computational capabilities, data structures, library subprograms, and input-output system. FORTRAN statements may be interspersed with CLIC statements in CLIC programs, CLIC programs may call FORTRAN subprograms, and FORTRAN programs may call CLIC programs. For the experienced user this access to FORTRAN allows him to make optimal use of the special features of CLIC, such as the question-answer logic, and he may still revert to FORTRAN when the CLIC language proves too limited.

The CLIC processing system also differs substantially from most other CAI systems. CLIC programs are translated into Fortran by a special preprocessor, after which the standard

Fortran compiler processes them to produce highly optimized machine code. A CLIC program may be saved in this compiled form and then executed whenever a student wishes to interact with the program. Execution of a CLIC program is extremely efficient both because of the compiled form of the program and because all data, including text, is stored in core storage along with the program so that the only input-output involved in execution is the input-output to the student's terminal.

Another advantage is clear: a partial solution to the widely recognized problem of inter-institutional system compatibility is at hand. If the institution has a FORTRAN compiler, transference and exchange of CLIC programs may be accomplished. Thus CAI capability may readily become available to a majority of the various computer systems.

The University of Texas Chemistry Project

In the past year, translation, revision, and expansion of the original CAI programs into CLIC has been accomplished. Utilization of the programs will commence in the fall, 1971.

Currently, the chemistry CAI facilities include access to a proposed 16-terminal "laboratory" to be used jointly with the physics, zoology, and computer science departments. In addition, 4 terminals are available solely for the chemistry project participants. The terminals are Model ASR-33 teletypes; access to the computer is over standard telephone lines. The terminals will possess random access slide capability for the display of information, spectra, lenghthy textual material, etc.

Following the original theme and strategies employed in the initial studies, the programs are supplemental to and coordinated with the conventional course material. Described broadly, the programs may be defined as either experiment simulation, remedial drill, tutorial dialogue, or evaluative interactions. Individualization is stressed through random function generation of numerical values, question sequence (in the drill programs), and extensive branching capability. In addition, most programs allow the student to request aid or additional information, skip, comment, or stop during a given interaction.

More specifically, the General Chemistry project will involve approximately 200 students in a total multi-media approach to teaching that will allow student access to film loops, audio-tutorial tapes, and the CAI programs of:

A. Tutorial/Drill

1. Scientific Notation
2. The Metric System
3. Temperature Conversions
4. Formula Writing
5. Inorganic Nomenclature
6. Percent Composition
7. Atomic Structure
8. Balancing Equations
9. The Mole Concept
10. The Gas Laws
11. Heat, Work and Energy
12. Energy Changes in Chemical Reactions
13. Stoichiometry
14. Concentration Units
15. Ionic Equilibria

B. Experiment Simulation

1. Millikan Oil Drop Experiment
2. J.J. Thompson Discharge Tube Experiment
3. Geiger-Marsden Experiment
4. Mosely X-Ray Experiment
5. Emission Spectroscopy
6. Colligative Properties

7. Acid-Base Titration
8. Calorimetry
9. Iron Ore Analysis
10. Kinetics
11. General Quantitative Unknown
12. Molar Volume of N_2
13. Qualitative Analysis, and
14. *Synthesis of cis-trans Isomers*

Approximately 150 students will participate in the Organic Chemistry project. Although the project does not currently make use of film loop or audio-tutorial teaching aids, fairly extensive supplementation of the course is provided via the CAI programs of:

A. Tutorial/Drill

1. Nomenclature of Alkanes
2. Nomenclature of Alkenes
3. Preparations and Reactions of Alkenes
4. Nomenclature of the Derivatives of Benzene
5. Preparations and Reactions of the Derivatives of Benzene
6. Organic Syntheses I
7. Organic Syntheses II
8. Nomenclature of Alcohols, Aldehydes and Ketones
9. Preparations and Reactions of Alcohols

10. Mechanisms:

    a. Halogenation of Alkanes

    b. Addition of HBr to Alkenes

    c. Dehydration of Alcohols

    d. Bromination of Toluene

    e. Bromination of Benzaldehyde

11. Elementary NMR Interpretations

B. Experiment Simulation

1. The Synthesis of 1-Methylcyclohexene and 3-Methylcyclohexene from 2-Methylcyclohexanol

2. Kinetic Studies I

3. Kinetic Studies II

4. Organic Qualitative Analysis

C. Evaluative

1. Quiz: General Organic Nomenclature

2. Quiz: General Organic Reactions

3. Quiz: Basic Reaction Mechanism

It is hoped that the expanded version of the chemistry project described above will experience the success exhibited by the earlier pilot studies.

References

1. Castleberry, S.J. and Lagowski, J.J. Individualized Instruction Using Computer Techniques. Journal of Chemical Education 47: 91-96; 1970.

2. Rodewald, L.B.; Culp, G.H.; and Lagowski, J.J. The Use of Computers in Organic Chemistry Instruction. Journal of Chemical Education 47:134-136; 1970.

3. Culp, G.H. An Approach to the Use of Computer-Assisted Instruction in Organic Chemistry. Doctor's Thesis. The University of Texas, Austin; 1969. Dissertation Abstracts: 30:5308-A, No. 11; 1970.

4. Castleberry, S.J. The Development and Evaluation of Computer-Assisted Instruction Programs on Selected Topics in Introductory College Chemistry. Doctor's Thesis. The University of Texas, Austin; 1970. Dissertation Abstracts: 31:1679-A, No. 3; 1970.

5. Castleberry, S.J.; Montague, E.J.; and Lagowski, J.J. Computer-Based Teaching Techniques in General Chemistry. Journal of Research in Science Teaching 7:197-208; 1970.

6. Culp, G.H. and Castleberry, S.J. Computer-Assisted Instruction in Undergraduate Organic Chemistry: An Evaluation of Selected Programs. Science Education. In the press.

7. Culp, G.H. and Lagowski, J.J. Studies Involving the Application of Computer Techniques to Undergraduate Organic Chemistry Instruction. Journal of Research in Science Teaching. Submitted.

MODERN METHODS OF HANDLING CHEMICAL INFORMATION
by Martha Williams

Chemical information and data are the products of testing, research, and development, and they become a resource for new or further research and development. Most chemists have an appreciation for, or associate a value with, information and data. They are also becoming increasingly aware of the fact that information today can be generated, processed, stored, managed, retrieved, and used in new ways. This is a result of the increase in use, and methods of economically using, computers in chemistry. Computers are used as tools in research. They are used for control and monitoring of experiments, and for simulating experiments--whether it be for analysis, kinetics, or synthesis of compounds. Once the research or testing is completed, the products of the work i.e., the data, can be selected, reduced from analog to digital form, tabulated, graphed, charted, printed or otherwise put in useable form. These computer-generated data--in either human-readable or machine-readable form, on paper, magnetic tape, disks, drums, chips or microform--now become a part of a vast, varied, and disperse data reservior.

Information and data can be considered to be of several types: (1) digitized numeric, as in the case of chemical handbooks, (2) analog, as in the case of IR or other spectral data, (3) symbolic, as in the case of structures of chemical compounds, and (4) alphabetic, as in the case of information expressed in natural language. These data, if they are to be used again as an input to further research, must be stored on an appropriate medium, arranged in a file structure suitable for searching, and managed intelligently. The existence of such a data base enables us to conduct searches and retrieve information that enables the researcher to obtain background information, keep up-to-date in his field, supplement the information he has, identify persons, organizations, documents, or other sources of information, or to make judgments as to whether the research he proposes has already been done.

Machine readable data bases have been developed as direct products of research and as secondary or tertiary products. That is, research may be directed toward generation of data--IR spectral for example--or, the results of research may be written up in journal articles that are then abstracted and indexed in secondary sources. The journal information or the abstract/index information may be put in machine-readable form either for purposes of searching or for purposes of automating production and printing functions. Prior to the advent of economic computer composition the cost of keypunching data to create large files for computer searching was very high and could seldom be justified on the basis of information retrieval alone.

Now, that large data bases are often created as a by-product of publication, the cost factor for input is no longer so significant. In addition to private files, many new data bases have become publicly available from commercial organizations, government agencies and scientific and technical societies. It behooves the chemist to become aware of these new sources of chemical information and to learn how to use them. He may use them within his own organization or he may purchase the service from information brokers in information centers.

During today's session we will hear reports of current activities in searching and retrieving chemical information and data. The papers and demonstrations will run the gamut from handling of analytical data, data for organic synthesis, symbolic data of organic structures, through natural language information in primary and secondary sources.

Conference on Computers in Chemical Education and Research
July 23, 1971
Northern Illinois University, DeKalb, Illinois

MODERN METHODS FOR SEARCHING THE CHEMICAL LITERATURE

Bart E. Holm
Secretary's Department
E. I. du Pont de Nemours & Co., Inc.

INTRODUCTION

Beilstein once said, "I read everything. I place it where it belongs." You can't do this any longer except in narrow fields. You are blessed or plagued with an abundance of published and unpublished information (Fig. 1). This is an historical continuation of a growing wealth of knowledge waiting to be put to use (Fig. 2). Chemists are fortunate, for of all the disciplines chemistry is unique with a tradition of information excellence and a tradition of information organization. Although you have growing information resources, you do have new and improving mechanisms for managing these resources. I will state the types of services a chemist requires and introduce the modern methods and formats used in these services. You will see the critical role played by computers. The papers and the demonstrations to follow will show, in detail, how these methods are being widely used to help the chemist. Many of these you can apply on your campus, many are inexpensive, all are useful. We hope you will see ways to advance the educational process through these methods and services.

CATEGORIES OF NEEDS AND INFORMATION

The information needs of chemists can be stated as the need for current awareness, selective dissemination, and retrospective search services, of research, development, engineering, production, and marketing information located internally or externally, and contained in journals, patents, theses, reports, data files, information services, and from people. Current awareness is continuous professional education through reading journals, attending meetings, and informal communications with colleagues. Selective dissemination is receiving information related to your interests from the remaining published literature beyond that scanned. Retrospective search is looking for information to answer a specific need. Chemists have traditional and new services in each of these categories.

Chemical information can be categorized as dealing with structures, data, and concepts.

Structures

Most of the effort in the information processing field has been in chemistry with much of that effort on improved ways of manipulating chemical structures. Chemists use chemical structures to communicate information about a chemical compound. Chemists need access to structural information, and the structure is a discreet entity amenable to organization.

Chemists commonly represent chemical compounds with pictures, an excellent method for one chemist to communicate with another using a blackboard. Each understands the meaning of the message. Unfortunately, the diagrams are not pronounceable, the diagrams cannot be ordered, and the multidimensional representations are difficult for printers (Fig. 3).

A number of nomenclature systems are in existence, each assigns one correct name for each compound. Nomenclature serves well for oral communication and for printing but nomenclature is not very effective for class searching because the main structural part carries many different names, depending upon the structure of the rest of the compound.

Beilstein Classification is a common way of identifying chemicals but it is difficult to search for subcode information unless you know the major classification.

Pictures, nomenclature, and classification have a place, but each has drawbacks for organizing large collections for indexing and searching. The major emphasis in recent years has been on fragmentation, notation, and topological coding (Fig. 4).

In fragmentation a compound is represented as a composite of its major structural features. Usually the fragments or functional groups are assigned numerical codes. Many schemes were developed in the late 1940's because the material could be coded on punched cards for later search. Most files have been converted to computer systems. Fragmentation is a useful technique but it does not define the structure completely enough for some searches. With notation the structure is represented by a single line of symbols. The next paper describes chemical notation systems. In topological coding the structure is represented by a unique array of symbols in a compact format for storage and search. The paper on the Chemical Abstracts Service includes mention of their registry file, chemical structures represented by topological coding. Some of the scheduled demonstrations show how these methods have been applied.

These three approaches are major advances enabling chemists to manipulate large collections of chemical structures to locate specific structures and to bring together groups of structures. The Committee on Chemical Information in 1969 published "Chemical Structure Information Handling - A Review of the Literature 1962-1968," publication 1733, National Academy of Sciences, Washington, 1969.

Data

Data are numeric or quantitative notations helping to describe a subject more precisely. Chemists are interested in melting points, boiling points, molecular weights, and specific gravity. These identifiers are easy to list and manipulate manually or by machine. One area, analytical data often includes instrumented techniques yielding indefinite curves or figures for compounds and mixtures. For identification these must be compared with the curves or figures of pure standards. Searching data by computer is included in another paper and in the demonstrations.

Concepts

Concepts cover qualitative information in chemistry as contrasted with data and structures. Concepts interact and overlap and are not mutually exclusive. Ablation, polymerization, extrusion, and spinning are all concepts and part of the chemical literature. Concepts have their own forms of presentation and manipulation. Here are some of the new approaches.

ACCESS TO QUALITATIVE INFORMATION

Current Awareness

Chemists read journals to learn the details of new developments and to keep current in their field. Most professionals do not try to scan all the journals that might have articles of interest. Those that try to scan many journals usually end up with tables in their office, den, or bedroom stacked with unread magazines. Subscribe to and regularly read a few core journals. Use a current awareness service to alert you to other material. Of the new services available two are representative. One called CURRENT CONTENTS®/Physical & Chemical Sciences, gives title information by reproducing the tables of contents of hundreds of foreign and domestic research journals. The weekly publication is extremely fast. The usefulness of the idea has caused many organizations to create their own table of contents publication to cover their journals in their area of interest.

The second service, Chemical Titles, listing titles of articles, is issued biweekly and covers 700 journals. Titles are rotated or permuted by computer to list the significant words in alphabetical order (Fig. 5). Since the work can be done by computer it offers efficient, inexpensive presentation. The title appears as many times as there are significant words in the title. A list of common words called a stop list is used to prevent alphabetizing on articles and other common words. These permuted listings are called KWIC (Keyword-in-Context). Some KWIC indexes drop the title information that cannot be included within a set number of characters for the line. An advance was made by taking the keyword out of the title and listing it on the left side of the page followed by the full title. These systems are called KWOC (Keyword-out-of-Context).

The next level of completeness for current awareness beyond scanning titles would be to receive abstracts. The total abstracts publications, Chemical Abstracts, Biological Abstracts, and Physics Abstracts are seldom used for current awareness today because of size. The approach by these organizations, is to produce subsets of the total in narrow fields. Chemical Abstracts is published in 80 sections. CAS also offers Basic Journal Abstracts, Chemical-Biological Activities, and Polymer Science and Technology. Index Chemicus, published weekly by the Institute for Scientific Information emphasizes synthesis, isolation, identification and/or biological activity of new chemical compounds. Other abstract services cover patents. These are all examples of published current awareness services.

Selective Dissemination of Information

Keeping up with the literature by current awareness techniques is often too big a task for an individual. Selective dissemination of information, usually called SDI, is the process of sending specific information to a person. Good librarians have done this for years. As they review the material coming into their collection, they send items to those known to be interested. People fortunate enough to have executive staff assistants have the same job performed for them. Now some services are available to help you and me.

Each person participating in an SDI service prepares a list of terms, called a profile, describing his area. As material comes in, the profiles are matched against the indexing terms, the titles, the authors, the references, or the abstracts to look for matching items. The person receives notification of newly received literature that matches his requirements.

In the most common form the notification consists of two punched cards. The first card contains a title and sometimes an abstract. The second card is a form to request the full article, and also gives a way to comment on how close the item matches the request. Systems adjust the profiles based on this feedback.

Some publishers offer SDI services to individuals or will provide magnetic tapes for processing by your organization for local SDI services. CAS offers magnetic tapes for Chemical Titles, Chemical-Biological Activities, CA-Condensates, and Polymer Science and Technology. Engineering Index offers a similar data base prepared from 3500 engineering publications. Excerpta Medica Foundation provides magnetic tape files covering 3000 biomedical journals. ASCA, Automatic Subject Citation Alert, from the Institute for Scientific Information processes profiles against the input to the Science Citation Index. The user supplies a list of authors or cited references and receives as a product all newly published work citing his stated references. In essence ASCA, or the Science Citation Index brings the researcher forward in time by listing those recently published documents which cite one of interest to him. In addition to authors and reference, profiles can consist of combinations of words, word stems, word phrases, organizations, and journals. One of the papers that follows describes how centers process data bases on magnetic tape to provide these services to you.

Current awareness and SDI services offer practical, economical, methods to assure that you are alerted to newly published work from all over the world. Read just a few core journals. Use current awareness and SDI for the rest.

Retrospective Search

You have heard how to keep current with the published literature. Some items you will read and discard; some you will keep. As soon as anyone accumulates a few thousand of anything, except money, he may have a problem finding what he needs when it is required. Here again, there are simple ways to organize your personal collections to find any item easily (Fig. 6).

First consider how you will search your files. Decide whether you will use single entry or multiple entry files. The admissions office keeps its records by the names of students. A similar system can be used in chemistry for a file on physical properties of chemical compounds if the only questions asked are to supply property values of specific compounds. In these cases pick your file subjects and set up the familiar classification system.

In science, frequently this does not occur. The need might be to locate all chemical compounds with a particular property value. A multiple entry system can handle multiple subject searches.

Traditional systems describe an item by a term or two. Index more deeply. Describe the articles in your collection by five to ten subject terms. Fully describe each item in order to provide flexibility in finding what you want quickly and economically.

The terms you select must be organized for consistent input and search. A modest amount of vocabulary control can assure that information is not lost because you indexed it one way and searched for it in another. Be consistent. Use CA nomenclature. The Condensed Chemical Dictionary is another good reference. If your subject area is extensive and complex consider formalizing your vocabulary by recording synonyms, generic levels, and related terms (Fig. 7).

A variety of storage forms are available for the index. First, give each article or item in your collection a sequential file number and store the items in order. Then set up a record for your index. Here is one form (Fig. 8). A card is used for each term. Record the item number on the term card that applies. In searching select the terms of interest and look for matching item numbers. In the example report 100 discusses "ablation of plastics in heat shields of space vehicles".

The Committee on Chemical Information has a file of over 300 articles on recent significant developments in chemical information processing. Abstracts for each are kept in serial order. The storage form for our index is a computer produced optical coincidence deck. Each card is a term. Item numbers are recorded as holes in the card and answers to questions are found by overlaying term cards.

Computers are used to produce indexes for manual searching as well as files for machine searching. Any of these methods can assure quick, complete retrieval from your files.

Microforms

Files can become voluminous. Collections of journals, abstracts publications, and patents can grow larger than the space available. Microforms have come into active use to reduce storage requirements, provide rapid access, and are an inexpensive way to provide multiple copies of whole documents. The common microforms are microfiche, aperture cards, and roll film in cartridges (Fig.9).

A microfiche is sheet microfilm about 4 x 6 inches commonly containing 5 rows of 12 images and a header strip for document identification. Government reports are available as microfiche for less cost than full size hardcopy.

Aperture cards are tabulating cards with a hole in which the film is placed. Normally one frame is included in the card although some systems hold up to eight frames. The Department of Defense stimulated the use with the ruling that all engineering drawings submitted for their contract work must be on aperture cards meeting their standards. U.S. patents are to be available on aperture cards.

The most familiar and economical microform is 16mm roll film. For high usage the film is stored in cartridges which simplifies threading in reading equipment. A 100 foot reel of 16mm film usually contains 2500-3000 8-1/2 x 11 inch pages of material. The American Chemical Society offers many of their publications in this form including ACS journals and Chemical Abstracts.

The thick listings you formerly received from the computer as printout can now be returned to you on 16mm film or microfiche by using COM (Computer Output Microfilm) equipment.

Chemical Information Resources

Chemical information is widely available in the United States. The American Chemical Society is active in providing primary and secondary services both as a wholesaler and a retailer. Of the commercial services in chemistry, the Institute for Scientific Information is the most extensive. A number of their products have been described and you will see others during the demonstration.

The BioSciences Information Service publishes a variety of products in their field including Biological Abstracts, and a KWIC index called Biological Abstracts Subjects in Context.

The National Library of Medicine in Washington is a reservoir of published literature in the medical sciences. Each month an abstract journal, Index Medicus, is published. All incoming journals are indexed for storage and retrieval for their computer system called MEDLARS.

The United States Government has many other information services of value to chemists. A notable resource is U.S. patents. The U.S. patent office has issued nearly 3.6 million patents, a rich source for chemists. The patent office has concentrated in recent years in improving their method for supplying copies of patents both as hardcopy and in microform. They have done little to improve the intellectual access to the content of patents. Commercial services fill this gap by offering abstracts and indexes to U.S. and foreign patents. In the United States, Information for Industry publishes the Uniterm Index to Chemical Patents and has magnetic tape services. In London, Derwent Publications inaugurated a Chemical Patents Index in January, 1970 providing abstracts, microfilm, and eventually indexes to worldwide chemical patents.

Summary

This has been an overview of new approaches to the processing of chemical information. Remember that a chemist needs a balance of current awareness, selective dissemination, and retrospective services. New techniques for handling structures, concepts, and data are available to help him in his work. New publications, formats, and services permit him to choose the inputs to his problem solving procedures. The computer is playing a key role in many of these services. These are all tools to help the chemist do his work more effectively (Fig. 10). The methods are available, many are inexpensive. They are widely used in industry and in the government. Can these methods help your faculty and your students, now and in the future? We are convinced that they can.

"ANOTHER BOOK? YOU HAVEN'T READ ALL YOU GOT NOW!"

FIGURE 1

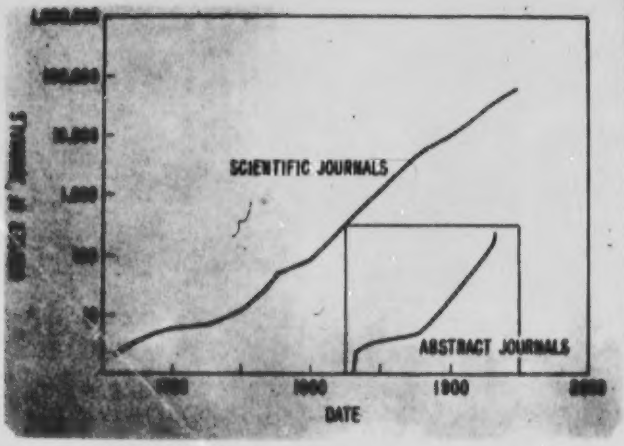
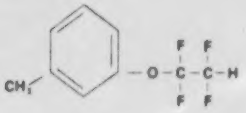


FIGURE 2

CHEMICAL STRUCTURES

Picture



Nomenclature

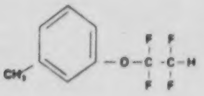
Phenetole, α, α, β, β, Tetrafluoro – m–Methyl

Beilstein Classification

526

FIGURE 3

CHEMICAL STRUCTURES



Fragmentation

Ether
Fluorine
Benzene Ring 292
1–1 Configuration
1–2 Configuration

Notation

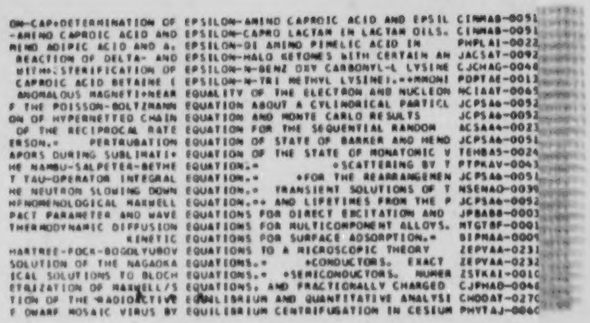
FVFXFFOR C

Topological Coding

C		
C	1	1
O	1	1
F	1	1
F	1	1
F	2	1
F	2	1
C	3	1
C	8	L
C	8	L
C	9	L
C	10	L
C	11	L
C	11	1
	12-13	L

FIGURE 4

15248

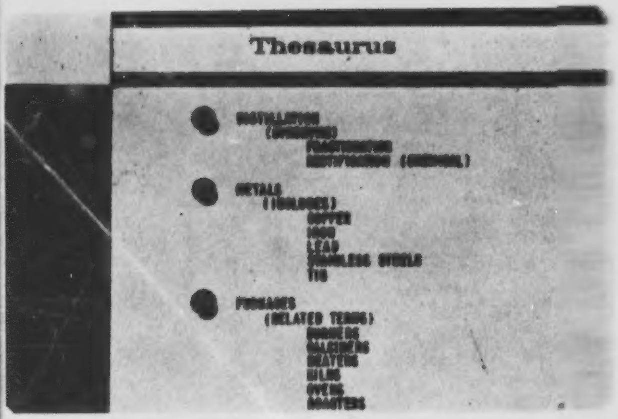


```
ON-CAP+DETERMINATION OF EPSILON-AMINO CAPROIC ACID AND EPSIL CINMAB-0051
-AMINO CAPROIC ACID AND EPSILON-CAPRO LACTAM IN LACTAM OILS. CINMAB-0051
MINO ADIPIC ACID AND A, EPSILON-DI AMINO PIMELIC ACID IN     PHPLAI-0022
 REACTION OF DELTA- AND EPSILON-HALO KETONES WITH CERTAIN AN JACSAT-0092
 WITH+:STERIFICATION OF EPSILON-N-BENZ OXY CARBONYL-L LYSINE CJCHAG-0046
 CAPROIC ACID BETAINE ( EPSILON-N-TRI METHYL LYSINE).*+AMMONI PDPTAE-0013
 ANOMALOUS MAGNETI+NEAR EQUALITY OF THE ELECTRON AND NUCLEON NCIAAT-0065
F THE POISSON-BOLTZMANN EQUATION ABOUT A CYLINDRICAL PARTICL JCPSA6-0052
ON OF HYPERNETTED CHAIN EQUATION AND MONTE CARLO RESULTS     JCPSA6-0052
 OF THE RECIPROCAL RATE EQUATION FOR THE SEQUENTIAL RANDOM   ACSAA4-0023
ERSON.*    PERTRUBATION EQUATION OF STATE OF BARKER AND HEND JCPSA6-0051
APORS DURING SUBLIMATI+ EQUATION OF THE STATE OF MONATOMIC V TEHBA5-0024
HE NAMBU-SALPETER-BETHE EQUATION.*          +SCATTERING BY T PTPKAV-0043
T TAU-OPERATOR INTEGRAL EQUATION.*     +FOR THE REARRANGEMEN JCPSA6-0051
HE NEUTRON SLOWING DOWN EQUATION.*  TRANSIENT SOLUTIONS OF T NSENAO-0039
HENOMENOLOGICAL MAXWELL EQUATION.*+ AND LIFETIMES FROM THE P JCPSA6-0052
PACT PARAMETER AND WAVE EQUATIONS FOR DIRECT EXCITATION AND  JPBAB8-0003
THERMODYNAMIC DIFFUSION EQUATIONS FOR MULTICOMPONENT ALLOYS. MTGTBF-0001
                KINETIC EQUATIONS FOR SURFACE ADSORPTION.*   BIPMAA-0009
HARTREE-FOCK-BOGOLYUBOV EQUATIONS TO A MICROSCOPIC THEORY    ZEPYAA-0231
SOLUTION OF THE NAGAOKA EQUATIONS.*      +CONDUCTORS.  EXACT ZEPYAA-0232
ICAL SOLUTIONS TO BLOCH EQUATIONS.*  +SEMICONDUCTORS.  NUMER ZSTKAI-0010
ETRIZATION OF MAXWELL/S EQUATIONS, AND FRACTIONALLY CHARGED  CJPHAD-0046
TION OF THE RADIOACTIVE EQUILIBRIUM AND QUANTITATIVE ANALYSI CHDDAT-0270
F DWARF MOSAIC VIRUS BY EQUILIBRIUM CENTRIFUGATION IN CESIUM PHYTAJ-0060
```

FIGURE 5

"NOW LET'S SEE —— "

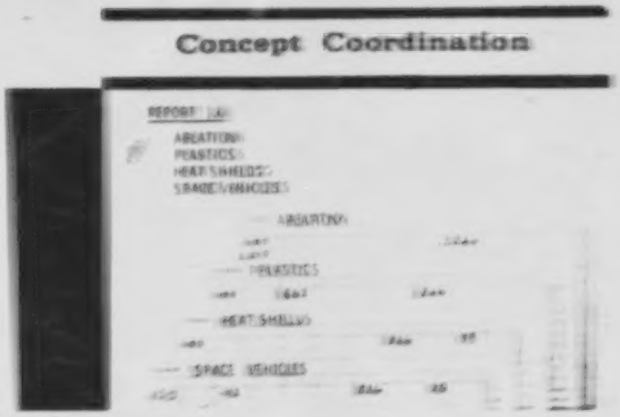
FIGURE 6



Thesaurus

- DISTILLATION
  - (SYNONYMS)
    - FRACTIONATION
    - RECTIFICATION (CHEMICAL)
- METALS
  - (INCLUDES)
    - COPPER
    - IRON
    - LEAD
    - STAINLESS STEELS
    - TIN
- FURNACES
  - (RELATED TERMS)
    - BURNERS
    - CALCINERS
    - HEATERS
    - KILNS
    - OVENS
    - ROASTERS

FIGURE 7



Concept Coordination

REPORT [illegible]

ABLATION
PLASTICS
HEAT SHIELDS
SPACE VEHICLES

ABLATION

PLASTICS

HEAT SHIELDS

SPACE VEHICLES

FIGURE 8

L 5-25

15.15

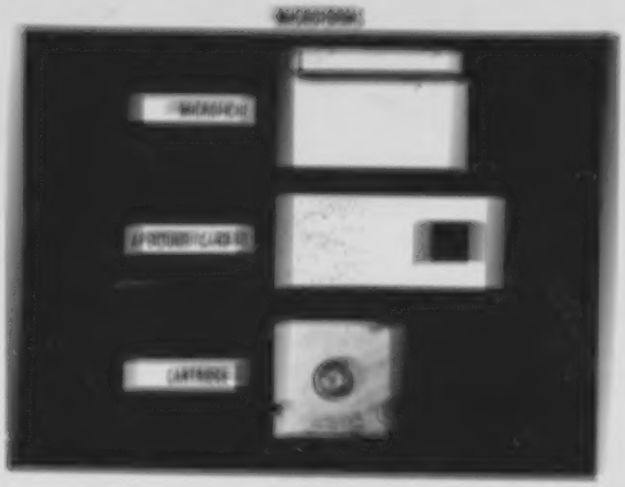

MICROFORMS
MICROFICHE
APERTURE CARDS
CARTRIDGE

FIGURE 9

FIGURE 10

10416

CHEMICAL NOTATION

A LINEAR DESCRIPTION OF CHEMICAL STRUCTURE

by

Carlos M. Bowman
The Dow Chemical Company
Midland, Michigan

10-15

ABSTRACT

Linear notation systems provide a machine-compatible description of chemical structures. The most widely used is the Wiswesser Line Notation (WLN). This report presents a brief summary of the WLN system, its background, applications, and significance in registry, searching, and correlative research.

For centuries chemists have found it necessary to devise systems to represent, in both written and oral form, the nature of chemical substances. First attempts were rather unsystematic and led to the development of trivial names. It is interesting to note, however, that as the deficiencies of trivial names become apparent, chemists started to use drawings to convey to other chemists the exact nature of particular chemical compounds. Thus, chemical structures or structural diagrams came into use. Over the years attempts have been made to systematize names and a large body of rules which govern the nomenclature of chemical compounds has gradually evolved. Nevertheless, chemical structures have continued to be a very important medium of communication between chemists.

We are talking primarily about methods by which chemical characteristics such as structure are represented on paper and orally. As compounds become more complicated, systematic nomenclature becomes very burdensome and chemists resort again to trivial names. However, when they want to express the actual structure, uniquely and unambiguously, they use structural diagrams.

In order to facilitate locating information about chemical compounds and classes of chemical compounds, chemists have organized names and structures into ordered lists which we commonly call indexes. In doing so, they have accentuated a particular component of the molecule, thus giving it prominence in their indexes. However, chemists have always wanted to be able to associate compounds with similar structure characteristics which are not necessarily those normally highlighted or accentuated in nomenclature and in ordinary indexing.

The advent of punch cards and later of computer technology prompted the development of coding schemes which permitted the ordering of names or structural diagrams or identification tags of compounds from a number of possible entry points. Perhaps the most widely used schemes are the so-called fragmentation codes. In using fragmentation codes a series of structures is broken up into significant components and codes are assigned to each one of these components or fragments. These codes are numbered or assigned particular punches on either regular computer type punch cards or edge notched cards. Many such systems have been developed, each one usually designed to meet the specific needs of the individuals using a particular collection of chemical compounds.

To retrieve information from one of these files, one merely has to select the fragments of interest, obtain the fragmentation codes assigned to those fragments, and then, through the use of a needle in the case of edge notched cards or a sorting or collating machine in the case of punched cards, go through the entire file and select those cards which have been coded with the desired fragments. This card retrieval usually completes the search, for each card contains a reproduction of the name, the structure of the compound, perhaps some of its basic properties, and usually a reference number that leads the user to files where more detailed information is available.

As the size of these coded files grew, it became imperative to devise faster and more definitive ways of retrieving information. With the advent of computers many of these systems were expanded and translated to computer searching so that searches could be done faster, and could accommodate more complex combinations of fragments. However, as we moved to computer searches of files which had been indexed through fragmentation codes we reduced the ability to retrieve immediately the names, the structures, and other information about compounds. All that we normally retrieve from a computer fragmentation type search is a listing of numbers which then necessitates reference, usually manual, to a separate file to obtain associated information. Another real problem with fragmentation codes is the fact that for a given compound there is not a unique set of codes, and from a particular set of codes it is not usually possible to reconstruct in an unambiguous manner the structure which generated that set of codes. Most fragmentation codes tell us what particular chemical fragments are present in the compound, but do not tell us much about how they are associated in a molecule. Thus, for large files the number of extraneous material which is retrieved is very significant and troublesome.

I think it is apparent then that chemists felt a need for the development of a code system which would permit the use of mechanical devices for storage and retrieval, which would permit the representation of a chemical structure in a unique manner, and which would allow the unambiguous reconstruction of the chemical structure from the code representation. In addition, the code should be precise and easy to use. If I were to list all the properties of such a system, you would very likely say that chemical nomenclature would meet most of these requirements, and that is true. However, nomenclature is not easy to use; that is, it is difficult to arrive at the proper name and once one has a name it is rather difficult to manipulate it using computers and other such mechanical devices. Thus, chemical notations came into being. Many notation systems have been developed over the years. G. Malcolm Dyson developed a notation which parallels chemical nomenclature(5); Herman Skolnik developed a system which is used at Hercules(10); other notation schemes include Hayward's(7), which has been studied at the U. S. Patent Office, and Silk's(9) in Europe.

It has been characteristic of these systems that they draw a fair amount of attention, and then fade away because of a lack of use. An exception to this has been the Wiswesser notation system, developed by William Wiswesser, which was first reported in 1954(16). I believe that this scheme of identifying chemical compounds would more than likely have gone the way of all the other notations had not a group of interested users decided to experiment with Wiswesser's ideas and concepts and make a concerted effort to prove and develop the system. Thus, over the period of the next 10-12 years, a number of chemists who were responsible for the retrieval of information from moderately large and complicated chemical files tried to use the notation scheme developed by Wiswesser. This resulted in a rather extensive revision which improved, clarified and extended the basic principles of the system.

Concurrently with the work of the information oriented people, a group of individuals interested in handling chemical structures via computers became interested in the notation and tried to write computer programs which would permit the detailed manipulation of codes derived from notations using the existing notation rules. It became apparent as the very systematic and rigid discipline of computer programming was applied to these rules that a number of inconsistencies and deficiencies existed. Consequently a cooperative effort among the original inventor of the notation, some of those people interested in information retrieval, and the computer people, led to the development of a manual of rules, published in 1968 as the Wiswesser Line-Formula Chemical Notation(12). We have since adopted the name Wiswesser Line Notation and use the acronym WLN. I will use that for the remainder of my talk.

I would like to emphasize that the present set of WLN rules are rules which have been tested and which have survived a fairly long period of use. They are not the rules conceived by one individual, but rather they represent the agreement of many people who have used the system in a real environment. Unlike many of the other notation systems which I have mentioned very briefly, earlier, the WLN has become widely used by many industrial concerns. In addition, one of the major information publishers, the Institute for Scientific Information, Inc., has launched a service, the Index Chemicus Registry System, based on the WLN system.

I would be the first to add, however, that the system is certainly not complete. It still has room for improvement and further development. Hopefully, my brief description today of the system and what can be done with it might interest some of our friends in the academic world to further test, develop and expand the applicability of the system.

The basic philosophy of the notation system is based on the use of a limited set of symbols to describe rather elementary components of a chemical structure. Listed on Figure 1 are some of the WLN symbols and the chemical characteristics which they represent. Because of time, I cannot really describe to you in great depth how the notation works. I can only give you some very simple examples of how it relates to chemical structures. The context-rich system uses only the 26 alphabetic characters, 10 Arabic digits and several special symbols which are commonly used, such as the asterisk, the dash, the ampersand, and the slash. The size of this limited vocabulary is essentially doubled by use of the blank space; symbols acquire a different meaning if preceded by a space. By the judicious assignment of these symbols to particular chemical characteristics, and by application of WLN rules, we are able to represent uniquely and unambiguously the majority of organic compounds using the Wiswesser Line Notation system.

On the next slides (Figures 2 - 6) I would like to show you some examples of chemical structures and the Wiswesser notations which have been derived for them. You will note that there is a one to one correspondence between the structural fragment and the notation symbol. In addition, the notation is a complete description of the compound inasmuch as connections between the various elements and groupings are clearly indicated. Since the notation is limited to commonly occurring symbols, it is easily adapted to computer manipulation and thus we have a simple way of representing in computer readable form the total structural information about a given compound or compounds.

There are many uses which have been made of the notation using computer systems. Perhaps the first application of computers to the handling of lists of

Wiswesser notations was the simple listing of notations in alphabetical order by Howard Bonnett(2). The nature of the rules and symbol assignment is such that an alphabetized listing tends to group together similar compounds. Thus some generic searching is possible from such lists.

A very logical extension to this was the application of Key-Word-In-Context techniques to notations to produce a permuted index(13). It is this type of index which the Index Chemicus Registry System produces on a monthly basis for new compounds reported in the literature(6). These two applications employ the computer to reorganize the notations and to produce a tool for manual retrieval.

Many other uses have been made of notations in the last five years and some very sophisticated searching techniques are now available, including some on-line conversational systems (1,15), one of which you will see demonstrated later this afternoon.

I have summarized on the next slide (Figure 7) some of the uses of WLN computer-based systems.

One of the more common uses of such a file is to ascertain the prior existence of a compound in a particular file. This can be accomplished with a relatively high degree of accuracy with notations; however, if one is primarily interested in registration and identification of unique compounds in a file, there are other methods, such as the Chemical Abstracts Connection Table system, which are superior for this particular purpose. Hyde and his co-workers at Imperial Chemicals Incorporated, Ltd. have successfully demonstrated the generation of structure displays from the notation(14); thus, an output from a search need not require additional lookup steps to find the name or require the scientist to learn the notation. Instead a picture can be presented to him directly.

The greatest use, of course, with notations has been for searching. Two general approaches have been used in this area. One has been direct searching of the notation as reported by Monsanto(15) and others. Usually in such a scheme a conventional text searching approach is used, in which a certain character or group of characters is identified and then looked for in the notation. This type of approach has not been too widely used because it is expensive and time consuming from a computer standpoint. A more commonly used approach is to design a computer program which generates fragments such as those discussed earlier in this paper, and then use the fragments to narrow down the size of the search file(4,8). If necessary, results of this coarse screening can be further searched by use of some kind of an automated on-line searching or by visually scanning lists of notations produced by the fragment search. It is possible to further extend the searching to an atom-by-atom type network match which would then produce complete specific retrieval. The economics of this last approach have not been attractive to date, and for that reason we have only seen some initial trials to establish the feasibility of such a concept.

Most notations can be decoded easily by a chemist with a brief training in the notation rules. Our experience has been that after one afternoon of training and a little practice, most organic chemists can read and interpret most notations. Coding is much more difficult if one is interested in the unique notation which conforms to the WLN rules. Training and extensive practice are necessary. However, the WLN rules are much easier to apply than nomenclature rules and coding costs are equivalent to connection table input costs. Algorithmic checking of rules of procedure can be applied to detect errors in encoding(3); however, the design of such programs is difficult and costly.

The economics of notation systems, however, do make them attractive for small to moderate size files. The ability to produce intelligible ordered lists, such as permuted indexes, permits extensive searching of significantly sized files. The generation of fragments has been another common use of notations. The complexity of the fragmentation schemes has varied from simple symbol fragments(1,11) to very extensive codes which can only be searched by computer systems(4,8).

In summary, the development of a chemical notation system for the linear representation of chemical structures has enabled us to store, in computer compatible form, complete descriptions of chemical structures. This capability has led to the generation of ordered lists and other manual searching tools for use in a conventional manner. It has also promoted the development and expansion of fragmentation-based systems which contain much more detailed and more consistent coding, thus enhancing a searching tool which has been in existence for several decades, and which has proven its worth time and time again. Finally, the notation system allows us to go to more detailed type searching. Since it is a complete description of the structure of a molecule, it also permits us to convert from one system to another. For example, with the proper algorithms we can go from a notation system to a connection table, which in turn can lead us to a systematic name.

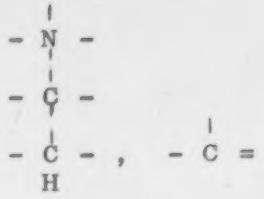
The availability of a complete, concise and precise description of a chemical molecule in computer manipulatable form opens up many avenues of work, including the potential for obtaining better understanding of the relationships between structure and properties of compounds. It is our hope that this brief description of a notation system and its possibilities will interest you sufficiently to learn more about it and to entice you into areas in which you can apply your interest in computer-based systems to studies of the manipulation of chemical structures and of structure/property relationships.

Figure 1.

WISWESSER LINE NOTATION

SELECTED SYMBOLS

B	Boron	E	Bromine
C	Carbon	G	Chlorine
F	Fluorine	M	Imino (-NH-)
N	Nitrogen	Q	Hydroxyl (-OH)
O	Oxygen	R	Benzene
P	Phosphorous	U	Unsaturation (=)
S	Sulfur	V	Carboxyl (>C=O)
-NA-	Sodium	W	Dioxygen (O= ⫽O)
-ZN-	Zinc	Z	Amino ($-NH_2$)



K $-\overset{|}{\underset{|}{N}}-$

X $-\overset{|}{\underset{|}{C}}-$

Y $-\overset{|}{\underset{H}{C}}-$, $-\overset{|}{C}=$

space&-/012...9ABC...Z*
(earliest latest)

Hierarchical Symbol Rank

Figure 2.

WISWESSER LINE NOTATION

Example 1.

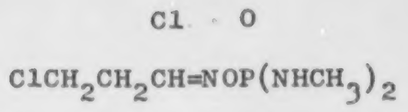
$CH_3CH_2CH_2OCH_2COOH$	SF
3 O 1 V Q	GF
QV1O3	WLN

SF	Structure Formula
GF	Graphic Formula
WLN	Wiswesser Line Notation

Figure 3.

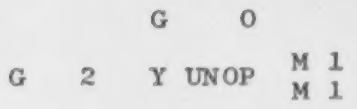
WISWESSER LINE NOTATION

Example 2.



$$\mathrm{ClCH_2CH_2\overset{Cl}{CH}{=}NO\overset{O}{P}(NHCH_3)_2}$$

SF



$$\mathrm{G\quad 2\quad \overset{G}{Y}\;UNO\overset{O}{P}\;\begin{matrix}M\,1\\ M\,1\end{matrix}}$$

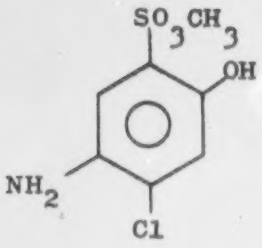
GF

G2YGUNOPO&M1&M1 WLN

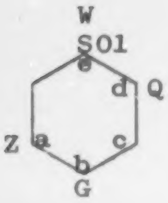
Figure 4.

WISWESSER LINE NOTATION

Example 3.


SO3CH3
OH
NH2
Cl

SF


W
SO1
e
d
Q
Z
a
c
b
G

GF

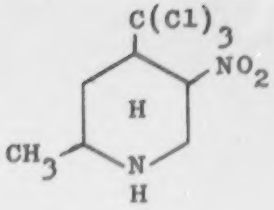
ZR DQ BG ESWO1

WLN

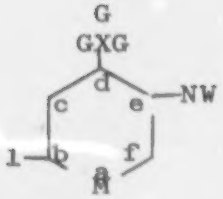
Figure 5.

WISWESSER NOTATION

Example 4.


C(Cl)$_3$
NO$_2$
H
CH$_3$
N
H

SF


G
GXG
d
c
e
NW
1
b
f
a
N

GF

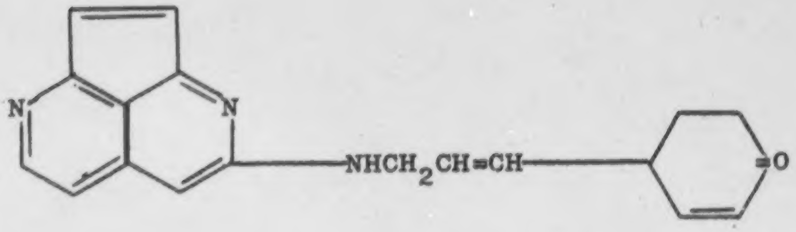
T6MTJ B DXGGG ENW

WLN

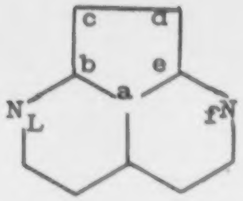
Figure 6.

WISWESSER LINE NOTATION

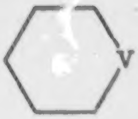
Example 5.


N
N
NHCH$_2$CH=CH
=O

SF



M 2 U 1



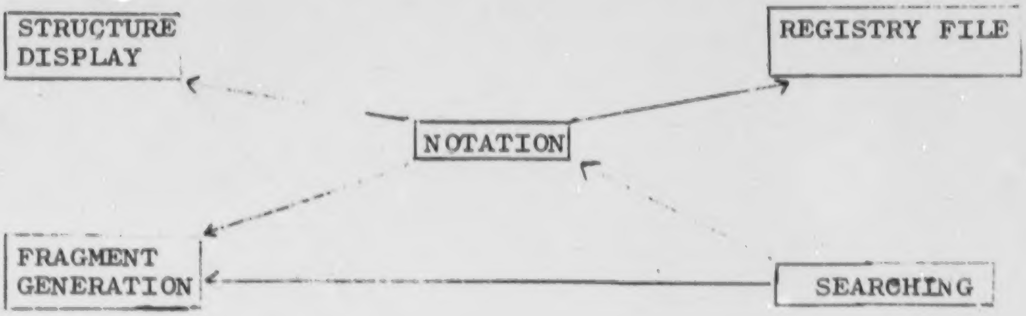
GF

T566 1A L FN LNJ GM2U1- DL6V BUTJ WLN

Figure 7.

WISWESSER NOTATION

COMPUTER BASED SYSTEMS


STRUCTURE DISPLAY
REGISTRY FILE
NOTATION
FRAGMENT GENERATION
SEARCHING

References

1. Bond, V.B., C. M. Bowman, N. L. Lee, D. R. Petersen, M. H. Reslock, "Interactive Searching of a Structure and Biological Activity File." Paper presented before the Division of Chemical Literature, 161st National Meeting of the American Chemical Society, Los Angeles, Calif., March 31, 1971.

2. Bonnett, H.T., D. W. Calhoun, "Application of a Line Formula Notation in an Index of Chemical Structures," J. Chem. Doc. 2, 2-6, 1962.

3. Bowman, C.M., F. A. Landee, M. H. Reslock, "A Chemically Oriented Information and Retrieval System. I. Storage and Verification of Structural Information," J. Chem. Doc. 7, 43-47, 1967.

4. Bowman, C.M., F. A. Landee, N. W. Lee, M. H. Reslock, B. P. Smith, "A Chemically Oriented Information and Retrieval System. III. Searching a Wiswesser Line Notation File," J. Chem. Doc. 10, 50-54, 1970.

5. Dyson, G.M., A New Notation and Enumeration System for Organic Compounds, Longmans, Green & Co., London, 1947. Second Edition, 1949.

6. Granito, C. E., M. Rosenberg, "The Chemical Sub-structure Index--A New Research Tool." Paper presented before the Division of Chemical Literature, 161st National Meeting of the American Chemical Society, Los Angeles, Calif., March 31, 1971.

7. Hayward, H.W., H.M.S. Sneed, J. H. Turnipseed, S. J. Tauber, "Some Experience with the Hayward Linear Notation System," J. Chem. Doc. 5, 183-189, 1965.

8. Hyde, E., F. W. Matthews, L. H. Thomson, W. J. Wiswesser, "Conversion of Wiswesser Notation to a Connectivity Matrix for Organic Compounds," J. Chem. Doc. 7, 200-204, 1967.

9. Silk, J. A., "A Linear Notation for Organic Compounds," J. Chem. Doc. 3, 189-195, 1963.

10. Skolnik, H., "A New Linear Notation System Based on Combinations of Carbon and Hydrogen," J. Heterocyclic Chem. 6, 689-695, 1969.

11. Smith, E.G., "Machine Sorting for Chemical Structures," Science 131, 142-146, 1960.

12. Smith, E.G., The Wiswesser Line-Formula Chemical Notation, McGraw Hill, New York, 1968.

13. Sorter, P.F., C. E. Granito, J. C. Gilmer, A. Gelberg, E. A. Metcalf, "Rapid Structure Searches Via Permuted Chemical Line-Notations," J. Chem. Doc. 4, 56-60, 1964.

14. Thomson, L.H., E. Hyde, F. W. Matthews,"Organic Search and Display Using a Connectivity Matrix Derived from Wiswesser Notation", J. Chem. Doc. 7, 204-209, 1967.

15. Waldo, W.H. et al. "Information Communication System," Monsanto Company, 1970. Private communication.

16. Wiswesser, W.J., A Line-Formula Chemical Notation, T. Y. Crowell Co., New York, 1954.

Machine-Readable Files from Chemical Abstracts Service *

by

Ralph E. O'Dette
Senior Staff Advisor
Chemical Abstracts Service

I will approach my topic from two points of view; first, the use at Chemical Abstracts Service of modern methods of handling chemical information and, second, some outputs of CAS that require modern methods for their exploitation. Both points will be illustrated by reference to the CAS Chemical Registry and the Integrated Subject File.

Modern methods must be defined at more than one level. Most apparent are the new devices and techniques that one can use to seek information from services that have been produced by means of other new techniques and devices. Of equal or greater significance, although perhaps less obvious, is the revolutionary change that modern methods can and should effect in the relation between an information user and the body of recorded information.

It is pertinent for information users to have some comprehension of the evolving CAS computer-based processing system because that system is radically different from traditional publishing procedures and technology. The computer-based system is far more flexible and, therefore, far more amenable to the changing demands of the user community than classical "hand operated" publishing could possibly be. In addition, modern technology makes possible the production of information files of radically new kinds.

---

* Summary of a presentation at The Conference on Computers in Education and Research, Northern Illinois University, DeKalb, Ill., July 23, 1971.

The CAS Chemical Registry is the nomenclature control system for CAS indexing of chemical substances. As such, the Registry is part of the day-to-day computer-based processing system that produces Chemical Abstracts, its indexes, and various specialized services. An integral part of the Registry is a continuously growing file of machine-readable representations of molecular structures which reflect all of the structural detail (e.g., stereochemistry, labelling, etc.) published by the author in referring to a substance. Associated with each substance record is a molecular formula and a unique, computer-assigned Registry Number which is a link to the CA Subject Indexes.

Information centers are learning to manipulate these files as search data bases to seek whole structures or any desired subset of atoms and bonds that may be less than a complete structure. The information-using community will soon be able to obtain search service from Registry files through information centers.

Newest CAS service is the Integrated Subject File, machine readable counterpart of the volume and collective Subject, Molecular Formula, and Registry Number Indexes to Chemical Abstracts. First made available to subscribers in ISF Volume 71 corresponding to the Volume Indexes to CA Volume 71, the 700,000 entries in ISF Volume 71 will occupy nine reels of tape in CAS's Standard Distribution Format. ISF Volumes 72 and 73 are expected before the end of 1971: all other CA volumes in the eighth collective Period (Volumes 66-75) will have ISF counterparts. The ISF is intended to be a continuing service.

5

Optimum utility of the ISF, probably the largest machine-readable information file to be made routinely available, will be determined by the user community itself which must be depended upon for development of search strategy, search software, and an overall philosophy of exploitation of this very large and very new kind of data base.

HANDLING OF VARIED DATA BASES IN AN INFORMATION CENTER ENVIRONMENT

by

Martha E. Williams
IIT Research Institute
10 West 35th Street
Chicago, Illinois 60616

Presented at the

Conference on Computers in
Chemical Education and Research
Northern Illinois University
DeKalb, Illinois
July 23, 1971

HANDLING OF VARIED DATA BASES
IN AN INFORMATION CENTER
ENVIRONMENT

DESIGN FOR UNPREDICTABLE FUTURE 10-38

- File Structure
- Preprocessors
- Hardware
- Software Modularity
- Machine and Installation Independence

USER ORIENTED DESIGN 10-44

- Data Base Options
- Profile Options
- Profile Features
  - Truncation
  - Logic
  - Linking or Grouping of Terms

USER AIDS 10-46

- Search Manual
- Truncation Guide
- Frequency Lists
- KLIC Index
- Bigram Frequency List

COMMUNICATION WITH USER 10-51

REASONS FOR USING THE SDI SERVICES OF AN INFORMATION CENTER 10-52

- Coverage
- Thoroughness of Search
- Consistency of Search
- Interdisciplinariness
- High Recall
- Cost-Effectiveness
- Speed and Regularity
- Timeliness
- Multiplicity of Data Bases
- Automatic Preparation of Files in Standardized Format
- Cost of Data Base Preparation and Operation of an SDI System vs. Subscriptions
  - Cost of Data Base Preparation
  - Cost of Operation of an SDI System
  - Cost of Subscription

ABSTRACT

The Information Sciences section of IIT Research Institute has designed and developed a Computer Search Center for processing and searching varied machine-readable data bases to supply information to a variety of users in industry, government, academic and other research institutions. The system was designed in a user oriented fashion and provides a full complement of options to the subscriber. These include: various data bases or portion of data bases; profile features such as inclusion or exclusion of any kind of data element that appears on a tape; full truncation capabilities on any kind of term; full free-form Boolean logic for resulting terms to one another; grouping of related terms; multiple sort options; and multiple print media options. User aids including Search Manuals, Truncation Guides, Frequency Lists, KLIC Indexes and Bigram Frequency Lists have been prepared for each data base. Twelve reasons why scientists should use the SDI services of information centers are discussed including cost benefits and comparative costs of data base preparation, in-house SDI, and subscription SDI.

DESIGN FOR UNPREDICTABLE FUTURE

Information centers exist to provide information from machine-readable data bases to users in industry, universities and other organizations. The Computer Search Center of IIT Research Institute was designed with a number of variables and uncertainties lying before us. We were not able to predict the data bases to be used, the character sets, the character codes, data base tape formats, data base record formats, data elements, and data base content. Nor could we predict the hardware we might use, user groups, user requirements, vocabulary, search requirements, output sorts, output formats, and output media. These are a few of the variables we saw. Our design had to accommodate all of them. We did not pre-determine which data bases and what user groups we would provide services to. We wanted to be in a position to handle any data base for which there is a significant market.

As we all know, search programs for handling machine-readable data bases are expensive to develop and expensive to maintain. We had no desire to incur the expense of maintaining multiple search programs. For this reason we developed a general purpose search program that would handle virtually any of the machine readable data bases containing natural language information.

When handling multiple data bases one is very likely to encounter multiple character sets and multiple character codes. The tape formats and record formats differ from data base to data base. In fact, they differ within data bases that are produced by the same organization. The data elements contained on the tapes vary considerably from tape to tape.

File Structure

In order to handle multiple data bases we had to determine a file structure that would accommodate all of the variables. The IITRI standard format does precisely that. With this format each record is composed of a key, directory, and character string. The key contains the volume, issue, and citation number as given by the data base supplier, and the directory identifies each type of element, contained in the record, according to IITRI data type codes. Some of the data type elements are listed.

DATA TYPE CODES	DATA ELEMENT
01	SOURCE INFORMATION --CODEN --JOURNAL REFERENCE --PAGINATION --DATES
02	TITLE OF ARTICLE
03	AUTHOR(S)
04	SHORT JOURNAL TITLE
05	KEYWORD(S) INDEX TERMS CA SECTION NUMBER
06	CA REGISTRY NUMBER
07	MOLECULAR FORMULA
08	CORPORATE AUTHOR
09	ABSTRACT TEXT
10	BA CROSS CODE
11	BA BIOSYSTEMATIC INDEX
12	EI CARD-A-LERT CODE
13	ORIGINAL LANGUAGE AVAILABILITY PUBLISHER PRICE DATE OF MEETING PARENT JOURNAL ORIGINAL ABSTRACT SOURCE

Figure 1. Data Elements and IITRI Data Type Codes

Key: 7416-081368 (Volume, Issue and Abstract Number)

Directory:

1	1	26	(CODEN)
4	27	14	(Journal)
2	41	76	(Author(s))
3	117	60	(Title)
8	177	51	(Corp. Author)
5	228	40	(Index Terms)
13	268	17	(Language)

String:

```
JPCHAX/75/3/325-30/000071/J.  PHYS. CHEM.VIBRONIC EFFECTS IN
THE INFRARED SPECTRUM OF THE ANION OF TETRACYANOETHYLENEDEVLIN,
J. PAUL$MOORE, JESSE C.$SMITH, DONALD$YOUHNE, YOUNG$DEP. CHEM.,
OKLAHOMA STATE UNIV., STILLWATER, OKLA.$CA073000$ IR SPECTRA
ALKALI METAL SALTSORIG. LANG.:  ENG
```

Complete Record Appears on Tape as:

```
7416-081368   1    1    26    4    27    14    2    41    76
3   117    60    8    177    51    5    228    40    13    268
17 JPCHAX/75/3/325-30/000071/J. PHYS. CHEM. VIBRONIC EFFECTS
IN THE INFRARED SPECTRUM OF THE ANION OF TETRACYANOETHYLENEDEVLIN,
J. PAUL$MOORE, JESSE C.$SMITH, DONALD$YOUHNE, YOUNG$DEP. CHEM.,
OKLAHOMA STATE UNIV., STILLWATER, OKLA.$CA073000$ IR SPECTRA
ALKALI METAL SALTSORIG. LANG.:  ENG
```

Figure 2. IITRI Formatted Citation

In the directory the code is followed by the starting position for the actual data, and an indication of the number of characters required by the data. Thus, in Figure 2, for the record having Citation Number 81368 of Volume 74, Issue 16, in Chemical Abstracts Condensates there is a CODEN which starts in position 1 and is 26 characters long. The next kind of data element included in the record is a Journal Name which is a type 4 code. The actual data starts in position 27 which is one position beyond the end of the CODEN data and is 14 characters long. The next data element is the Author Name, which is a type 2 data code, and starts in position 41 which is one position beyond the end of the journal data. Author data is 76 characters long, and on down the line. If you follow through Figure 2, the format becomes obvious. The string portion of Figure 2 shows how the actual data for this particular reference is contained in IITRI format on tape, and the complete record which appears in the lower portion of Figure 2 shows the entire key directory and character string for the particular record as it appears on tape.

The use of data element codes allows us to handle multiple varied data elements. The system also allows us to add new data elements and new data type codes as they arise. We have no way of knowing what new data elements suppliers may include in their tapes a few years from now. However, we have allowed for 9,999 different data type codes. It is unlikely that we will be unable to accomodate any new data element that should come into existence.

The standard IITRI format is employed for any data base processed. Our method for handling multiple data bases is to write a pre-processor program for each different data base that is handled in the system. The pre-processor program reformats the data that is contained on the supplier data base and puts it into IITRI format. In that way every data base looks the same to the search program, and all data bases can be handled by one and the same search program.

Preprocessors

The cost of writing pre-processor programs is very slight compared to the cost of maintaining multiple search programs. A preprocessor or format conversion program takes about 2 man weeks of programming effort to write. We initiated this system of format conversion for a generalized search program three years ago. It has worked very well. It has allowed us to accommodate changes in data bases whenever the need arose. Fortunately or unfortunately the data base suppliers do make frequent changes in the format and content of their data bases. These changes may not be large or terribly significant, however, as far as computer programs are concerned, any change can be disastrous if one is not able to accommodate it.

The use of preprocessor together with a flexible and expandable file structure, have made it possible to accommodate multiple data bases with varying contents, varying data elements, differing character sets and character codes, and different tape formats and record formats.

Hardware

The next area of unpredictability that affected our design is related to hardware. We, like many organizations, have changed computers with some regularity. We did not want to be locked to any particular configuration, nor to develop our software in an assembly language that would be very machine and installation oriented. The system had to be as machine independent and installation independent as possible. For this reason we decided to adopt a higher level compiler language and to use a family of computers that enjoyed wide acceptance throughout the country and internationally--IBM 360. The compiler language we adopted was PL/1, a language that is highly amenable to handling natural language data. PL/1 compilers are available on IBM 360 series computers and they have been announced for Burroughs and CDC hardware.

Software Modularity

We wrote the programs in a modular fashion in order to easily accommodate changes to any portion of the system. The principal modules in the systems include: (1) format conversion for reformatting data bases; (2) a profile editing module, that handles the terms, logic and user related information that are associated with each profile; (3) a search module which is the heart of the system and matches user profile terms against terms on the data base within the requirements of the profile; (4) a formatting module for preparing output in dissemination format; (5) a module that sorts and prints the output; and (6) a statistics gathering module.

Machine and Installation Independence

The reasons for developing our software in a machine independent and installation independent fashion were that we anticipated our own hardware might change--which it did--, we wanted to be able to install our system at other organizations that had a need for running SDI within their own facilities, and we wanted to be able to provide profile-writing training courses and workshops at other locations. Successful achievement of machine and installation independence with IBM equipment is evident from the fact that we have run our programs at nine different computer facilities with different: hardware, computer models, versions of the operating system, peripherals, and releases of the PL/1 compiler. Figure 3 indicates the varieties we encountered in the nine facilities. In no case did we have any real difficulty. Preparation of appropriate JCL is usually all that is required.

Hardware:	IBM 360	Models:	40 50 65 67 75
	IBM 370	Model:	155
	Any computer with PL/1 Compiler		
Processors:			MFT MVT PCP HASP
Operating System Versions:			15-16 17 18 19 19.6 20
PL/1 Compiler Releases:			4.1 5 5.2

Figure 3. Machine and Installation Independence of IITRI Software

USER ORIENTED DESIGN

Data Base Options

The current awareness or SDI (Selective Dissemination of Information) system has been operational since the spring of 1969. Data bases handled by the Computer Search Center (CSC) are Chemical Abstracts Condensates, Biological Abstracts, BioResearch Index, and Engineering Index's COMPENDEX. Two more data bases are planned for the near future and other data bases will be added depending on user needs. We will handle any data base for which there are enough subscribers. Searches are conducted and output sent to users weekly, biweekly or monthly in accordance with the frequency of the particular data base to be searched.

Profile Options

One may include searchable elements as positive or negative search terms, i.e., one may require the presence or absence of any particular search term to qualify a citation as a "hit" citation. Among the searchable elements are those shown in Figure 1. The search terms may be single words, multi-word terms, phrases or portions of words, i.e., any legitimate character string. The range of options available to users for their profiles is shown in Figure 4.

Profile Features

The principal features built into the system to achieve effective profiles are the following: All forms of term truncation; full free form Boolean logic with any degree of nesting; grouping or linking of similar terms; and weighting of terms according to user assignment of relevance. Weighting is used to indicate relative importance of terms, separate closely related concepts and to sort output in relevance order.

Truncation

The terms themselves may be truncated to facilitate retrieval of terms containing common fragments. The four truncation modes allowed are shown in Figure 5. The usefulness of right truncation is usually readily understood. Right truncation is used to select singular, plural and other forms of words that contain a common stem. The usefulness of left truncation is not so obvious but it can be readily demonstrated. For example one might use the left truncated term *MYCIN to represent antibiotics and retrieve:

Terms - anything other than single character

- Single word
- Multi word
- Phrase
- Fraction of term
- Symbol or acronym

Kinds of Terms - anything on the data base

- Author
- Company Name
- Country
- Language
- Date
- CODEN
- BA Cross Code
- BA Biosystematic Index
- Keywords/Index terms
- CA section number
- CA registry number (when available)
- Molecular formula (when available)
- EI Card-a-lert Code
- Abstract or Text term
- Any term on any data base searched

Sorting of Output

- Author
- Weight
- Citation number

Output Media

- 5" x 8" cards
- Paper
- Multilith masters
- Tape
- Microfilm

Figure 4. Profile Options

Mode	Function	Example
none	requires exact match of a term	term AZO
left	allows substitution of any prefix on the term	* term DI AZO
right	allows substitution of any suffix on the term	term * AZO XY
both	allows substitution of any prefix and/or suffix	* term * DI AZO METHANE

NOTE: * denotes truncation

Figure 5. Truncation Modes

ACTOMYCIN
ANTIMYCIN
BIOMYCIN
ERYTHROMYCIN
NEOMYCIN
STAPHYLOMYCIN
STREPTOMYCIN

The one search term *MYCIN substitutes for 20 to 30 specific terms which are not shown. The use of simultaneous left and right truncation would pick up all of the above terms plus plural forms e.g., "STREPTOMYCINS", and those with terminal punctuation e.g., "ACTINMYCIN.".

Logic

Profile terms are related to one another by means of logic symbols. The logic symbols used for the logical operators AND, OR, and NOT are:

Logical Operators	Symbol
AND	&
OR	|
NOT	¬

The logic expressions for profiles can be as specific and involved as is necessary to express the user's question. While most expressions are relatively simple, any expression can be handled by the system. For example, the following expression would be legitimate:

(((A&B) | (C|D|E|F)) &¬G) | ((H&I) &¬J)

Linking or Grouping of Terms

In order to simplify the writing of a profile, similar terms may be linked together by a link code and then the group will be referred to by the single letter code. An example is given in Figure 6, where a user has requested information on reactions of halogens and alkali metals.

User Aids

In addition to providing many options for preparing profiles we have developed a number of aids to help users write profiles, select words, truncate words, and word fractions. Index Terms and Hit Terms are printed on each output card to provide the user with information for revising his profile.

Terms	Link Code		Terms	Link Code
Halogen	A		Alkali metals	B
Halide	A		Lithium	B
Fluorine	A	AND	Sodium	B
Chlorine	A		Cesium	B
Bromine	A		Potassium	B
Iodine	A		Rubidium	B

Rather than:
(Halogen | Halide | Fluorine | Chlorine | Bromine | Iodine)
& (Alkali metals | Lithium | Sodium | Cesium | Potassium | rubidium)

User specifies: (A&B)

Figure 6 - Linking or Grouping of Terms in a Profile

Search Manual

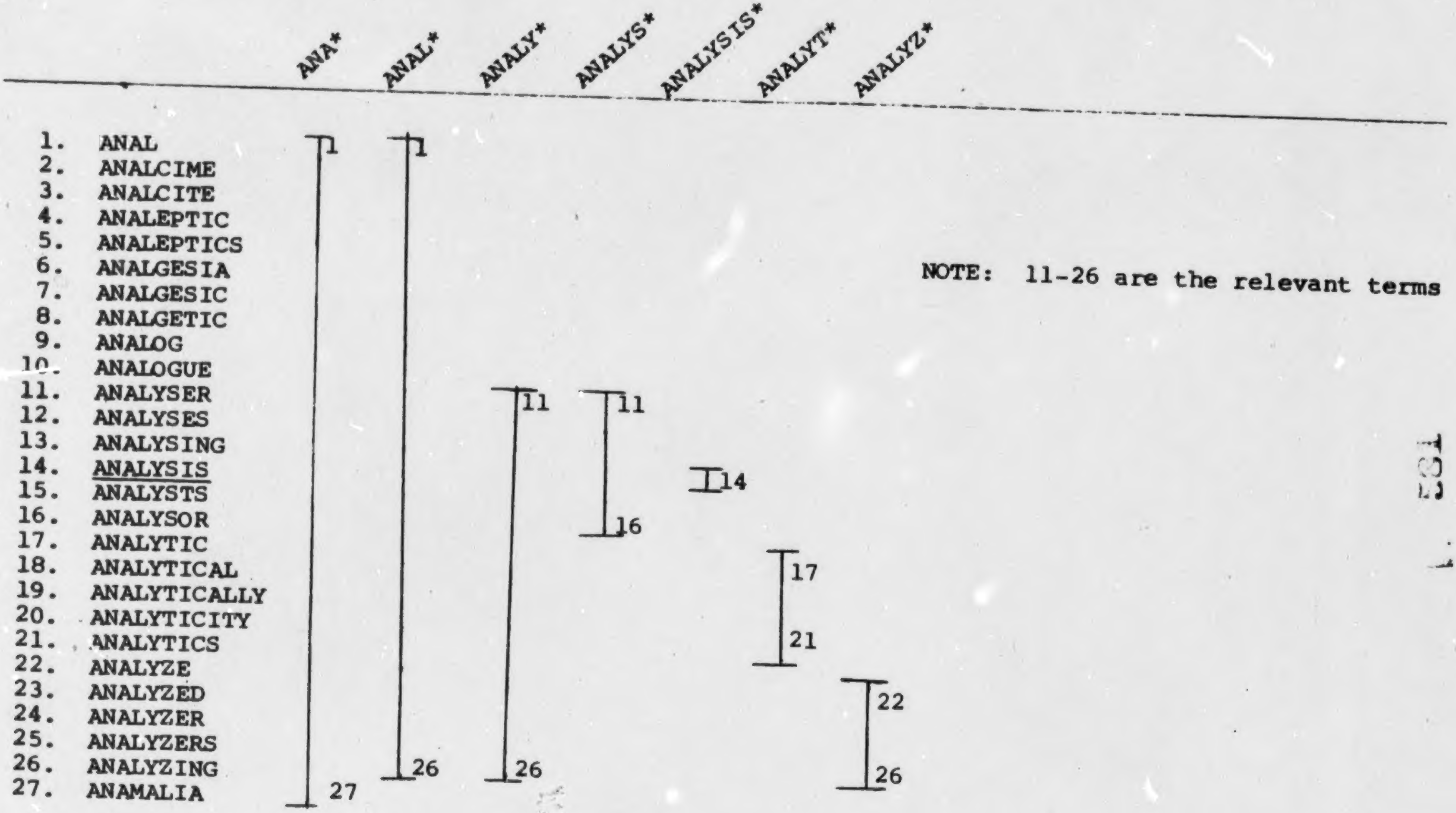
The CSC Search Manual explains the basic techniques of profile writing, including: question formulation, concept identification, concept expansion, profile refinement and profile modification. Methods for using search terms, truncation, logic, links and weighting are described. A Supplemental Guide has been written for each data base. The guide demonstrates profile writing tailored to the specific data base.

Truncation Guide

A Truncation Guide illustrates where to truncate a term in order to retrieve the maximum relevant words with the minimum noise. Figure 7 from the profile Truncation Guide demonstrates the retrieval ability of various forms of terms related to the concept ANALYSIS.

Frequency Lists

A Frequency List in frequency order and a Frequency List in alphabetic order have been prepared for each data base. They are used to assist in selection of search terms. A high frequency term will produce a high volume of hits unless it is combined with another search term or assigned a low weight. A low frequency word might be used independently. Frequency lists are used as rough indicators of the volume of output one might expect to receive for specific terms. Our Frequency Lists have been prepared for one volume of each data base.


ANA*
ANAL*
ANALY*
ANALYS*
ANALYSIS*
ANALYT*
ANALYZ*
1
27
1
26
11
26
11
16
14
17
21
22
26

1. ANAL
2. ANALCIME
3. ANALCITE
4. ANALEPTIC
5. ANALEPTICS
6. ANALGESIA
7. ANALGESIC
8. ANALGETIC
9. ANALOG
10. ANALOGUE
11. ANALYSER
12. ANALYSES
13. ANALYSING
14. ANALYSIS
15. ANALYSTS
16. ANALYSOR
17. ANALYTIC
18. ANALYTICAL
19. ANALYTICALLY
20. ANALYTICITY
21. ANALYTICS
22. ANALYZE
23. ANALYZED
24. ANALYZER
25. ANALYZERS
26. ANALYZING
27. ANAMALIA

NOTE: 11-26 are the relevant terms

Figure 7. Truncation Guide Entries For The Concept ANALYSIS

KLIC Index

A KLIC Index (Key Letter-In-Context Index) has been prepared for each data base. The KLIC Index is a lexicographic ordering of all terms in a data base by each character (alpha, numeric or special) in the term or character string. It is a permuted arrangement sorted by character with the balance of the term wrapped around it. See Figure 8. The KLIC is used for linguistic research and as a user aid. By consulting the KLIC one can determine the retrieval capability of a particular letter combination or term fragment. The KLIC is used to identify letter combinations that are highly specific and would therefore be discriminating search terms e.g., the character string *YBD* does not occur anywhere in the CA or BA data bases except in the term MOLYBDENUM (Note: in a literary data base it would occur in the mythological character SCYLLA and CHARYBDIS). Thus, *YBD* could be used as a search term for molybdenum. On the other hand, letter combinations that occur frequently in many irrelevant terms should be avoided e.g., the letters RNA for ribonucleic acid could be used as a search term assuming one did not specify simultaneous left and right truncation *RNA*. The simultaneous truncation mode would retrieve more than 200 irrelevant words. Some of these are:

ALTERNATE

BARNACLE

CARNATION

DIURNAL

FINGERNAIL

MATERNAL

Bigram Frequency List

The KLIC Index indicates where letter combinations occur and our Bigram Frequency List, which provides a frequency count for every two letter combination in the data base, indicates how often each bigram occurs.

```
                      KEY-LETTER-IN-CONTEXT INDEX

---------- |-------------    ---------- |------------
      ANIM ALS' /                  ASPH ALT /
     CRYST ALS' /                 DICOB ALT /
     MINER ALS' /                 ROCKS ALT /
     ORBIT ALS' /                HIGH-S ALT /
    CHEMIC ALS' /                SULFOS ALT /
    MATERI ALS' /                TRICOB ALT /
TION-RADIC ALS' /CA             ACYLCOB ALT /
           ALSACE /             FUSED-S ALT /
/          ALSACIENNES          WATER-S ALT /
         B ALSAM /             IRON-COB ALT /
         B ALSAMINA /          MOLTEN-S ALT /
         V ALSARIA /           RADIOCOB ALT /
         S ALSBURY /           ZINC-COB ALT /
         F ALSE /             AMMINECOB ALT /
       CAT ALSE /             AQUEOUS-S ALT /
         W ALSER, /           ORGANOCOB ALT /
         S ALSES-LA /         SULFATE-S ALT /
         W ALSH /            COPPER-COB ALT /
         W ALSH, /           IRON(II)-S ALT /
        TH ALSIMINE /        NICKEL-COB ALT /
 DIHYDROTH ALSIMINE /        ANINATOCOB ALT /AL
           ALSINCIDES /      ARBIDE-COB ALT /C
H20 /   NA ALSI04-SI02-      NICKEL-COB ALT /IRON-
           ALSI06 /          YRAZINECOB ALT /P
    PERIST ALSIS /           LADIUM-COB ALT /PAL
        LI ALSI206 /         ANGANATE/S ALT /PERM
        CA ALSI208 /         ATINUM-COB ALT /PL
        NA ALSI308 /         CHEIDEANST ALT /S
    CZOCHR ALSKI /           MARIUM-COB ALT /SA
LER-NAPIER ALSKI /BISCH      RIALLYLCOB ALT /T
      MICH ALSKI, /          TTRIUM-COB ALT /Y
      TEMP ALSKI, /                   S ALT. /
     KATCH ALSKI, /          /      COB ALT(III)UREA
    CZOCHR ALSKI'S /         YANIDE)COB ALT(I) /ISOC
         P ALSKY, /          LT(II)/COB ALT(II /COBA
         W ALSMANN, /               COB ALT(II) /
         C ALSNS /            OXIDE-COB ALT(II) /
           ALSO /            YRAZINECOB ALT(II) /P
         S ALSOLA /           LIGANDCOB ALT(II). /
         S ALSOLINE /        PTIDE /COB ALT(II)-DIPE
           ALSON /           AMIC / COB ALT(II)-GLUT
      INGW ALSON, /          ILOTR /COB ALT(II)-NITR
         H ALSTEDII /        ,4'', /COB ALT(II)-4,4'
           ALSTEN, /         LT(II /COB ALT(II)/COBA
/          ALSTONERINE,             COB ALT(II), /
/          ALSTONERINES      AAMMINECOB ALT(III /HEX
           ALSTONIA /        TAMMINECOB ALT(III /PEN
         R ALSTONITE /              COB ALT(III) /
ES /       ALSTOPHYLLIN         AQUOCOB ALT(III) /
         G ALSTYAN, /               COB ALT(III). /
         V ALSUGANA /        ICALL /COB ALT(III)-OPT
    NAPHTH ALSULFONIC /      MOTED /COB ALT(III)-PRO
         M ALT /                    COB ALT(III), /
         S ALT /                      S ALT) /
       BAS ALT /                    COB ALT) /
       COB ALT /                   ANST ALT) /
```

Figure 8. KLIC Index Entries

Communication with User

In order to maintain good rapport with users and to be sure that their profiles are functioning efficiently to provide the desired information, CSC uses many avenues of communication with users. Among them are:

- Unlimited profile changes
- Low cost profile switch
- Evaluation reports (95% response implies use)
- Feedback cards
- Continuous precision calculations
- Telephone contact
- Comments on profiles to suggest changes in logic weighting, and grouping of terms, or to suggest use of new data elements or new terminology

The concern for users is of extreme importance to information centers. Information systems are designed to be used and if the clients are not satisfied with the service they will not use it.

REASONS FOR USING THE SDI SERVICES OF AN INFORMATION CENTER

There are some scientists that have no need for or cannot use SDI. There are those who have no need to monitor or use the new developments in a field, those for whom there is no appropriate data base, those whose area of interest is so extremely narrow and highly specialized that the appropriate terminology is unlikely to be used by authors in titles or by indexers in index terms, and those who can truly monitor the publications in a field by scanning a handleable number of journals. The members of this group are not nearly so numerous as they might think.

It remains, then that there are many scientists who do have a need for SDI, and there are advantages and reasons for using the SDI services of an information center. They are: (1) the extensiveness and inclusiveness of the broad coverage, (2) thoroughness of search; (3) interdisciplinariness; (4) high recall; (5) cost effectiveness; (6) speed and regularity; (7) timeliness; (8) multiplicity of data bases; (9) automatic preparation of files in standardized format; and (10) cost of data base preparation and operation of an SDI system vs. subscriptions.

Coverage

The first and most obvious reason for SDI, is that for the most part, it simply cannot be done manually any more because the volume of publications is so large. While formerly one could "eyeball" Chemical Abstracts or other secondary sources to cover the literature in his area of specialization, this would now be a monumental task. The journal coverage of data bases such as CAS Condensates, BA Previews and others is quite extensive; for example, in the field of chemistry a year's worth of Condensates includes 300,000 chemical references taken from approximately 20,000 journals -- such extensive coverage cannot be duplicated elsewhere. In the field of biology, BA Previews includes approximately 250,000 references selected from approximately 8,000 journals. In engineering the COMPENDEX data base produced by Engineering Index includes 75,000 references a year from 3,500 journals. There are many additional data bases providing extensive coverage in their respective fields.

Thoroughness of Search

Machine searching is more thorough than human searching. The computer will search every profile term against every citation on the data base - no reference is overlooked and human fatigue is not a factor. Once the term matches have been made then the "hit" citations are checked to see if they satisfy the logic the user employed to relate the terms to each other.

Consistency of Search

A machine search evaluates every citation in the data base in exactly the same manner. The search strategy and criteria for selection are employed in the same way for the first citation on a tape and for the last citation. While a human searcher is likely to be affected by fatigue and boredom, the computer is not.

Interdisciplinariness

One of the most important reasons for using a machine search is related to the interdisciplinary character of research. The user may be working in a field that requires coverage of several subfields within a data base or several data bases. For this reason it would be very difficult to identify and manually search all the appropriate portions of a data base. It is also becoming increasingly difficult to anticipate which area of subspecialization a particular journal article might be assigned to within a secondary source. For example, an article dealing with a particular air pollutant might find its way into any one of a number of sections in CA -- all of which would be correct.

An organic chemical could be assigned to one of the organic chemistry sections; if detected by some analytical device it could be attributed to one of the analytical chemistry sections; as an air pollutant, to the air pollution section; if it were inhaled and produced a biologic effect it might be assigned to toxicology; if it were deposited on a body of water it would apply to water pollution; and if deposited on the ground it might apply to the section on soil and plant growth.

This is just one example but there are many and this interdisciplinary factor has been cited by our users repeatedly as being a real advantage. Many users, who had manually searched CA for years, thought they were doing a good job, and thought they knew exactly which sections of CA would be appropriate, were very surprised to find that the SDI search located relevant references from sections they would never have examined.

High Recall

Because of the previously mentioned advantages it is possible for the user to achieve a higher recall with SDI than he could manually. Naturally, the value of high recall vs. high precision varies from user to user depending on his objective in using the service. The user or the profile coordinator acting for the user, must weigh the tradeoffs and determine whether he can afford to miss a few relevant references in order to achieve high precision or whether he can afford to retrieve some irrelevant references in order to ensure his not missing anything.

An example of the recall ability of the system is the case of one of our user companies that maintained a manual search system parallel to the SDI we provided and compared the output for over a year. For many years and throughout the experimental year 20 bench chemists divided up the sections of CA and searched for references relevant to the companies areas of research. The output of their search was forwarded to their technical library as was the output of our SDI. Eleven profiles were written and run against Condensates. The results of the study was that the manual search identified 5% more relevant references than the machine and the machine identified 15% the chemists missed. Simply, if total relevant references identified by both sources is considered 100% then the recall for the manual search by professional chemists was 87.5% and the recall of the SDI system was 95.8%. Naturally, the SDI produced about 60% non-relevant citations but the time required to evaluate and reject these was not significant. The truly significant factor is the economic one.

Cost-Effectiveness

The value of an SDI system can be measured in terms of time saved. There are many other values but cost effectiveness is the criteria that is most often applied by the subscriber. Not all cases are so dramatic, but in the example cited above the cost of the manual search using average rates of $20,000/man year was $87,000 whereas the cost for the eleven profiles was $2,800 (or $4,500 using our current price schedule where subscription rates are related to number of profile terms).

An American Chemical Society survey reported in Chemical and Engineering News 47:3;, July 28, 1969 that the average industrial chemist spends 11.8 hours per week in current awareness and literature searching. Of the 7.5 hours spent on current awareness SDI effected an average savings of 3.1 hours for every hour spent. Assuming an expense to the company of $15.00/hours the savings over a year would be almost $2,500. Perhaps much of the time would have been spent in off hours and would not have netted the company additional productive man hours. However, conservatively speaking, a rule of thumb might be that if the user saves only one hour per week, at $15.00/hours the cost is $780.00 which is considerably more than the average cost of $250/year for an SDI profile.

Speed and Regularity

A machine search is done very rapidly and the search for each user is done each week on a regular basis. The machine does not have time off for illness and vacation. The human searcher on the other hand, may become burdened with other tasks and may not have time to go to the library each week. Even in cases where the users area of research is narrow enough and well defined so that he can readily do his own current awareness of CA by searching a limited and manageable number of sections the question remains, will he do it and will he do it on a regular basis? If the library is located some distance from his office or if he is one of many users on a long distribution list, he is unlikely to read the current issue when it is published. SDI provides him his output on a regular basis regardless of these circumstances. At IITRI, we have been providing SDI from CA Condensates on a production basis for two years and we have never missed a weekly run due to anything related to our system. (In one instance the supplier was late in producing the tape causing a delay in the search).

Timeliness

The magnetic tape version of a secondary source is usually made available prior to the publication of the hard copy. At IITRI the Condensates search tape is received 1-2 weeks prior to publication of the Chemical Abstracts. The tape is held for a week to provide a backup copy in case any issue is deleted in production or in the U.S. mail. The search is conducted and output received by users approximately 1-2 days prior to receipt of the hard copy in their library. This enables the user to check abstracts at the time he reviews his output citation cards.

Multiplicity of Data Bases

Many organizations require journal coverage from a variety of sources in widely divergent disciplines. They need to use several data bases or secondary sources and if this is done in-house they must be aware of the coverage, terminology and indexing practices of the different suppliers. And, as changes in the data base occur they must be accommodated by the search in preparing search terms and strategies.

Automatic Preparation of Files in Standardized Format

The output of an SDI search is citations that are printed in a standardized format. In cases, such as our system at IITRI, the citations are printed, together with index terms and hit terms, on cards thus providing a unit record file from which irrelevant or obsolete material can be deleted. In contrast with printouts on computer paper the citations can be sorted and filed according to the user's preference and it is not necessary to retain irrelevant citations. The user's file can be purged and rearranged as needed.

Even in cases where a user can search the current issues of CA or another data base it might be worth it to subscribe to SDI solely for the convenience of having the appropriate references printed on cards in standard format. Few scientists relish copying citations out of CA nor is it always convenient for a secretary to go to the library to type them.

Cost of Data Base Preparation and Operation of an SDI System vs. Subscriptions

Cost of Data Base Preparation

The cost of preparing a data base, which is borne by the suppliers and incurred only once, is high. The information center and subsequently the subscriber pays only a small fraction of the cost.

The Condensates data base covers approximately 20,000 primary journals, includes more than 25,000 new citations per month, and the cost of preparing this data base in machine readable form is very high. It is unlikely that an individual company or other user organization could afford to select, abstract, edit, index, keypunch, compose, print and otherwise prepare such a collection. Even preparing a data base from the journals known to be of interest to an individual company would probably cost much more than the total subscription fees required to provide SDI from the same sources.

Clearly, data base preparation is an expensive task and user organizations would not want to duplicate the efforts of the data base suppliers. The next alternative is that of processing data bases in-house and operating an SDI system. This too, is more expensive than one might anticipate. The mere purchase or lease of a data base is not all that is required to run an SDI system. In fact, the costs associated with obtaining a data base are extremely small relative to the balance of the expenses.

Cost of Operation of an SDI System

Operating costs can be considered as four in kind: data base; materials, equipment and furniture; machine time; and personnel. Details for some of the specific cost items are given in Figure 9.

Data Base Related Costs:

Purchase or lease of data base

Royalty payment to data base supplier for all hits or citations disseminated

Materials, Equipment and Furniture:

Purchase or lease of keypunch and terminals if needed

Purchase of expendable materials e.g. cards, paper products, postage, office supplies and various out-of-pocket expenditures including travel

Furniture e.g.: special file cabinets to hold tapes, printouts, listings, etc., and standard office furniture.

Machine Time:

- Production:
  - Reformat data base
  - Edit profiles
  - Prepare profile input for search
  - Search
  - Sort and format output
  - Print
- Statistics:
  - Tape library maintenance
  - Research and Development

Personnel Time:

- Management
- Marketing
- Systems design
- Programming
- Profile maintenance
- Keypunch
- Clerical

Figure 9. Center Costs

The efficient operation of a center requires (1) A management component to direct and oversee all production, research and development activities; (2) A marketing component to develop a market and ensure use of the center (3) A systems designer is required where software systems are to be designed in house--this type of staff is not essential where available software is used (4) Programming capabilities are required to: (a) develop software (b) modify existing software for internal use, (c) change and improve software, (d) adapt to changes in operating system, compiler and configuration and (e) adapt to changes imposed on the center by data base suppliers such as new data elements, changes in data base format, changes in record format, changes in machine code, changes in storage densities, etc. (5) A subject specialist is needed to handle profiles and maintain liaison with users.

Profile coordination involves writing profiles; monitoring profile performance, output and user response; keeping abreast of data base changes such as indexing procedures, vocabulary, subject and journal coverages etc., updating and modifying profiles in response to changes in user interest or in data bases; and maintaining user records with respect to precision values, number of hits and any other type of statistic the center may wish to maintain.

The data base expenditures and materials are the smallest items in a center's budget. In our own case they amount to less than five percent of our expenses. Computer time and personnel time are the major cost elements. Some people have been rushed into thinking that for the relatively minimal investment of $5,000 to $10,000 for acquisition of data bases they can process tapes in-house or operate centers. This is a very unrealistic outlook. A rule of thumb might be that efficient processing of a data base would require at least 100-200 users and an information center needs about $100,000 per year to operate. This is a broad generalization, since I fully realize that software efficiences and personnel requirements will vary among centers depending on the overall system, computer installation, quality of profiles, requirements of users and services provided. Bear in mind that these costs assume that software is already available and need not be developed.

Cost of Subscription

The expense involved in purchasing subscriptions to the SDI services of information centers is by far the most economical way of providing SDI to the average company. Individual profiles range in cost from approximately $100 to $500 per year depending on the center, the data base, the number of terms and the volume of output. The average company can purchase many subscriptions to many data bases for a small fraction of the cost required to process in-house. Use of existing centers provides the user organization with far more flexibility in terms of varieties of data bases, types of services, etc. than could be experienced in-house. The user company has no commitment to a hardware configuration or software package, nor does he have a payroll to meet.

The number of centers in operation in the U.S. today is more than adequate to meet the needs of the limited market. They and their sponsors have borne the cost and headaches associated with design and development, and they are now ready to share the fruits of these efforts by providing service to industry, academic institutions and government facilities. For the most part, development costs are not passed on to users in subscription fees -- only operational costs -- so it behooves members of the scientific community to take advantage of the investment.

A NEW APPROACH TO COMPUTER ASSISTED DESIGN OF ORGANIC SYNTHESES

W. Todd Wipke
Department of Chemistry
Princeton University
Princeton, New Jersey 08540

"In other words, automated planning of synthetic routes could enhance progress by five-fold and develop the state of the art to the same high level achieved in the elucidation of structure. It seems quite reasonable to anticipate that computer automation will eventually come to synthetic chemistry just as new instrumentation has come to structural problems.

To be perfectly fair, it must be admitted that such automation cannot be expected to materialize soon. The prodigious problem of programming stands as a barrier which will be assailed only where the incentive is equally great."[1]

This paper will higlight the important aspects of this "prodigious problem of programming" paying particular attention to the issue of representation. Part of the material presented here is drawn from the first synthesis program, OCSS, by the author and E.J. Corey,[2] part from LHASA, which is OCSS with a considerably extended chemical module by the author, E.J. Corey, R.D. Cramer,III, and W.J. Howe.[3] The remainder pertains to the new synthesis program being developed at Princeton by the author, T.M. Dyott, and P. Gund.

A View of the Problem

Designing a synthesis is a challenge for a chemist because it focuses all of organic chemistry toward the solution of one problem. In the same way, it is a challenge to teach a computer how to plan syntheses, because it brings to bear not only the field of organic chemistry, but also that of computer science toward the solution of one central problem.

By building such a program we gain problem solving capability, we gain understanding of our own problem solving processes in general and that of synthetic analysis in particular, and perhaps most important, we gain new ideas for laboratory research. Anyone who has programmed a computer knows that in programming a solution to a problem, one develops a complete understanding of the problem. In removing the mysteries from synthesis we must be careful not to simply transfer them to the area of computer science. That is, we must also remove the mysteries of programming complex problem solvers.

Synthesis can be viewed from several different directions depending in part on the motivation for the synthesis. The first type might be characterized as an industrial point of view, given A, what can be made from it with a given range of reagents in a few steps. By necessity

Type I $A \xrightarrow{?} ?$
Type II $A \xrightarrow{?} T$
Type III $? \xrightarrow{?} T$

this problem requires working forward. A type II synthesis is one characteristic of very complex molecules such as progesterone. Here the target T is given and a similar complex available material A is chosen as starting material. The problem is to find the interconnecting paths between them. Finally there is the type III synthesis, characteristic of total syntheses. The target is specified, but there is no prejudice as to reactions or starting materials. Of course this problem requires working backward for a solution. We are currently most concerned with this latter approach, but methods developed for the type III will have some applicability for the other types as well.

The general theory of synthetic analysis as applied to complex molecules has been previously discussed[2] and a particularly simple yet powerful technique for the design of organic syntheses has been termed the "logic-centered" approach. In its "pure" form the method starts with the synthetic objective or "target molecule", T, and by synthetic inference derives the set of precursor molecules which can be expected to be converted to the target molecule by one synthetic step or a sequence of simple steps. Repetition of this process treating each precursor as a new target generates the familiar "tree" of synthetic intermediates. An important goal of the "logic-centered" approach is to lead to precursors which are simpler than the target. The analysis terminates upon production of precursors which are acceptable as starting materials.

The "logic-centered" approach while it lacks the exciting intuitive insights of the chemist is still quite powerful and is well suited for computer execution. The first synthesis program, OCSS (Organic Chemical Synthesis Simulation), used these principles. The validity of the synthetic tree produced by OCSS is a true reflection of the effectiveness of the logic-centered principles, for a computer cannot as yet rely on "prior synthetic experience."

The Computer's Role is Obvious, or is it?

It sounds so simple, "feed the computer all known reactions and let it apply them in every possible way." This brute-force approach fails even with the biggest computer, because the number of possible branches in the synthesis tree is so large. It can be argued that synthesis is more complex than chess and it has been estimated that there are 10^{120} different paths in a complete chess maze.[4] Clearly the brute-force method won't do.

The clue to intelligent behavior is selective search guided by heuristic rules. One such rule has already been mentioned--the striving for simplicity. Thus transformations which do not ultimately simplify the target are pruned as early as possible. Good chemical heuristics minimize the ever present risk of pruning away good solutions.

In theory the computer could solve the entire synthesis problem without human intervention, but in practice we have insufficient knowledge of chemical reactivity and of proper evaluation criteria to permit a closed solution. To close the loop we use a chemist to evaluate results and guide further search. By saving some of our human chemical input for execution time, we gain flexibility and power to meet situations unanticipated during the programming. Programming the closed solution is hampered also because of the difficulties in fitting such large problems to current computer memories and of organizing and controlling it once there. The chemist-computer combination for now appears to be the best arrangement.

The General Scheme

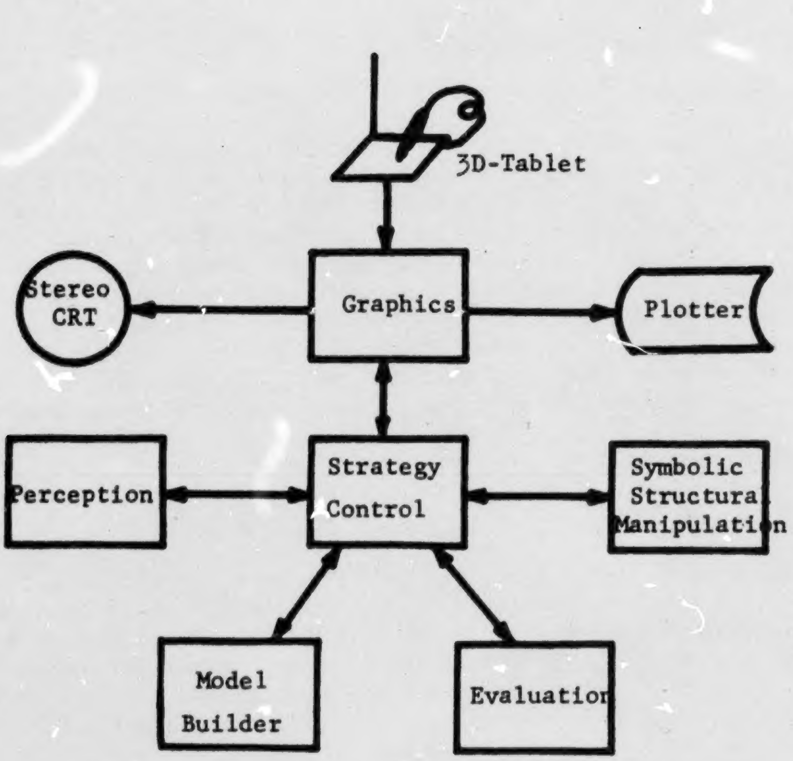
The elementary problem solving scheme shown below is applicable to the problem of synthesis regardless of direction of analysis, forward or backward.

1. Communication of problem
2. Perception of problem
3. Selection of chemical operators
4. Creation of sub-problems (precursors or products)
5. Evaluation and output of result, Quit if done
6. Selection of next sub-problem and go to 2.

Fortunately, the above functions interact infrequently and in well defined ways. This enables a modular program design and the use of overlay structure to increase the effective size of the computer. The program being developed at Princeton embodies a new approach of giving the computer a molecular model to which it may refer in its analysis. Thus where the OCSS family of programs had five modules, there are now six modules:

1. Graphical communication
2. Strategy and Control
3. Model builder
4. Perception of structural features
5. Symbolic structural manipulation
6. Evaluation of precursors

Several of these modules will now be described to portray in a general way the form of information and processing involved and the significance of the form.



Graphical Communication

The graphical communication module lets the chemist-computer team converse freely in terms of standard structural diagrams. OCSS pioneered the use of a two-dimensional Rand Tablet and cathode ray tube(CRT) for this purpose. Recently we reported preliminary work using an acoustic three-dimensional tablet for entry of chemical structures into a computer and an Evans and Sutherland display system for stereo viewing of the structures.[5] The acoustic tablet consists of three strip microphones and a pen which emits small sparks. With this equipment which was developed at Princeton, the program can sense the pen X,Y,Z coordinates and the status of a button on the side of the pen which signifies the beginning of a bond or a pointing action. The current graphics module is written using a new FORTRAN General Interactive Graphics Language[6](GIGL) and permits entering two-dimensional structural diagrams, freehand three-dimensional drawings, or a relatively accurate molecular model traced from an actual physical model of the Fieser or Dreiding type.

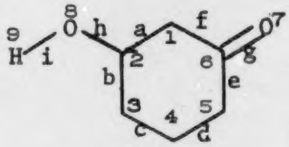
By using graphical communication, debugging is simplified, the language and nomenclature problems are avoided, and the program becomes palatable to organic chemists who now can understand the input and output without learning codes or in the case of 3-D, without having to supply atomic coordinates. The internal representation which the chemist never sees is a rather standard connection table which forms the fundamental structural representation.[2]

Model Building

This module must construct a fairly accurate three-dimensional model of the target molecule and subsequent precursors if requested, regardless of the type of structure, with or without initial atomic coordinates and in a minimum amout of time. The approach taken incorporates heuristic classical-mechanical constraint satisfaction techniques[7] to derive a model of minimum "strain energy." It relies on the chemist to guide it to the desired conformation of the molecule, which he can easily do using the acoustic tablet in 3-D. Experience with the program indicates that it can usually build a reasonable model given only a connectivity table or even given random coordinates--showing that it is good at avoiding false minima. Details of this program module will be presented in a later paper.

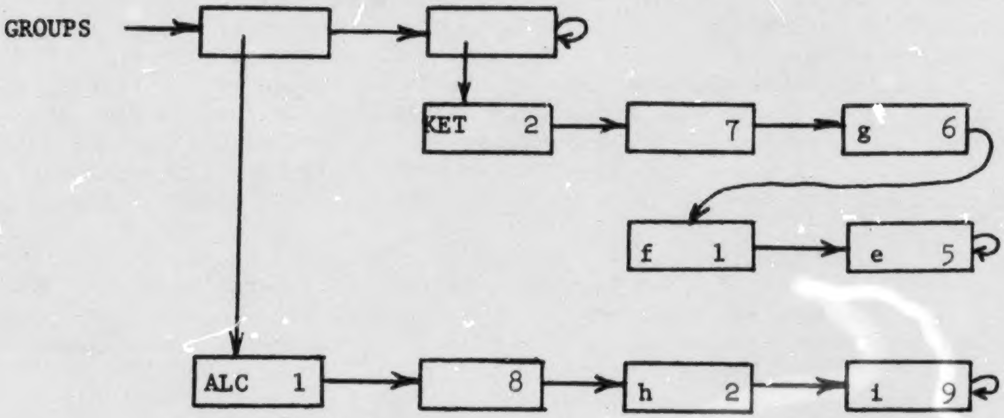
Perception of Chemical Features

The perception module, the eye of the program, takes the raw connection table created by the graphics module and digests it into more useable and recognizable forms. This processing is carried out on a variety of levels from primitive to complex, creating new data structures appropriate to the complexity of the perception. On the lowest level information is simply transferred from the connection table into sets. A set is a contiguous string of binary digits (bits). If the ith bit in the set is a "1", then the ith object is a member of that set. Examples are given below. Although sets can store only binary information, the combination of sets by logical operators provides a simple means for expressing complex processing. Further, digital computers process bit strings in parallel, 32 or 36 bits at a time, making set operations the most efficient processing available. A complete set of FORTRAN routines provides every conceivable set operation.



SAAA(2)	= 101000010	Set of atoms alpha to atom 2
SADA(3,2)	= 000010000	Set of atoms degree 3 away from 2
SAASA(SAAA(2))	= 111101011	Set atoms α to α set of atom 2
OXYGEN	= 000000110	Set of oxygen atoms
BOND2SET	= 000001100	Set of doubly bonded atoms
AND(OXYGEN,BOND2SET)	= 000000100	Set of doubly bonded oxygen atoms

On the next level of complexity, functional groups are recognized.[2] Now we need to store the type of functional group, and how the "standard form" of the group maps onto the target molecule, that is, the origin of the group and its constituent atoms and bonds. This requires a new data structure called a list or linked-list. A list is composed of cells which come from and are returned to a common storage pool of cells. The "head" of the cell contains a datum or a pointer to a sublist and the "tail" contains a pointer to the next cell on the list or an "end" code. The groups in the previous example would be represented as below. (The box represents the head of the cell which contains packed data.)



Each sublist is a group which has a type and a group sequence number. Cells following contain a bond-atom pair except for the first cell. The chemist's origin of the group is usually the second cell of the group (the third cell of the sublist). Lists are useful for storing data which is of variable length and structure such as groups and rings. For the flexibility, we pay a penalty of a 50% increase in consumption of storage space and processing time.

The perception continues, building on lower level perceptions, to recognize and represent more complicated groupings such as aromatic rings and groups extended by congugation with double bonds. Lastly perception utilizes its molecular model and derives chemically significant features such as close spatial arrangement of two groups.

Control and Strategy Module

Preliminary decisions on what general goals should be sought are made by the strategy module. Using the perceived information and any global information, it attempts to determine what the most important problem is, eg, to change foreign ring systems into familiar ones, to change sensitive groupings into more stable ones, etc. It also keeps track of the precursors created by subgoals so they can be operated upon by the higher level goals. The design of the goal oriented executive was based on early artificial intelligence research in chess,[8] and geometry theorem proving.[9] The generality of this technique permitted major changes in the synthetic strategies and processing with little or no change in the executive.

Transform Selection or "What to Do"

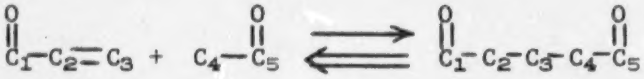
Everything prior to this has been preparatory for this stage of processing. The problem is "how do we choose what chemical transform to use?" Each chemical reaction can be characterized by the pattern of functional groups or other structural features it leaves in the product. Thus we organize the reactions by special features or relationships found typically in the products of the reaction. For example, the Aldol reaction leaves a 1,3 hydroxy-ketone or 1,3 double bond-ketone in the product. The first step then in answering this question is to make appropriate associations between target molecule and chemical reactions.

Functional groups are assigned to classes according to functions they perform, eg, CYANO group is also an electron withdrawing group (WGROUP). Two groups are selected and the relationships between them are determined as the number of bonds separating them, stereochemical relationships or even spatial relationships. Each relationship with all possible pairings of class names is matched against the associative memory. (This is a simplification, there are several associative tables.) A match means the relationship may be chemically significant. So far the search has been target-driven.

When a match is found in the associative memory, an interpreter takes over in a table-driven search to evaluate the environment of the groups and the path between them. The interpreter examines coded entries in a chemical table which cause it to determine the presence or absence of certain features and increase or decrease the rating of that transform accordingly. The idea for this target-driven:table-driven combination stems from the design of the efficient functional group recognizer in OCSS.[2] This combination only requires storage of the key structural features recognized by the general perception module, and leaves detailed perception to the environment evaluating interpreter as needed to answer specific queries in the table. It would be impractical and undesirable to attempt to anticipate all possible queries by perceiving all possible relationships.

But perhaps the most important feature of this method, and in fact the reason it was selected, is the language used to create the coded tables. This language was developed for OCSS in 1969 and since then has undergone considerable extension. A simple associative entry for the Michael reaction is written in an early dialect of the language to demonstrate the readability of the statements.

A key concept in organic chemistry is that bonds are made along a linear path on the structure although the path may be cyclic. Thus when the program finds a pair of groups and the path between them, it has already established a reference system for describing and executing a transformation involving those groups. A canonical ordering of the groups determines where the numbering starts.



$$\overset{O}{\overset{\|}{C_1}}-C_2=C_3 + C_4-\overset{O}{\overset{\|}{C_5}} \rightleftharpoons \overset{O}{\overset{\|}{C_1}}-C_2-C_3-C_4-\overset{O}{\overset{\|}{C_5}}$$

Michael Reaction. Single arrow is the synthetic direction; double arrow is the analytic direction. The partial associative entry is shown below.

```
OXO OXO 5
PRIORITY 50
SUBT 30 IF WGROUP CBN3
SUBT 10 FOREACH RGRP ALPHATO CBN3 OFFPATH
KILL IF DBOND ANYWHERE ONPATH
      .
      .
      .
CONDN SLALK
BREAK BOND3
MAKE BOND2
```

When a match is found in the associative memory, the matched transformation is given a base priority for execution. The environment of the groups is checked by qualification statements in the associative entry. In the example, the presence of an electron withdrawing group on C_3 on the path lowers the priority of the transform by 30. The priority of the transform is lowered further by 10 for each alkyl group attached to C_3 not on the path. Finally, the transformation is dropped from consideration if there is a double bond anywhere on the path C_1-C_5. If there are groups sensitive to slightly alkaline conditions, the conditions of this reaction, then the priority is further lowered. The last two entries direct the program how to perform the transformation in the analytic direction. The transforms are executed in the order of their final priorities with a minimum priority cut-off.

Evaluation

Machine evaluation of precursors has so far only been concerned with the valence of atoms, electronic stability, implausible bonding, and in a crude way with precursor simplicity.[2] The molecular model will improve the evaluation of strain in the structure. The missing criterion, which the chemist has supplied, is the liklihood of the forward reaction occurring as predicted. A forward working chemistry module could certainly help fill this gap.

Remaining Problems

The OCSS program has been quite successful in the design of syntheses of molecules containing relatively few interacting functional groups. A major problem is the ordering of operations on multifunctional compounds. In part this is a result of the difficulty in predicting relative reactivity of complicated functional groups.

Another problem is that of writing machine independent programs. Our current project is using ASA FORTRAN IV only because it is universal. The language is poor for symbolic processing. We have however through the "plex" programming technique[10] managed to make the program independent of the exact machine representation of the data even though the data is packed. Thus transfer from one machine to another requires rewriting only a small machine language interface.

Conclusions

The future of this problem solving method appears bright. The new approach of having a 3-D model may open the areas of stereo-selectivity, steric hindrance and proximity effects. One can envision giving the program an electronic model as well with which to evaluate charge distribution and orbital symmetry. Certainly the power of the programs will increase as its factual chemistry base increases and as chemical heuristics are developed. While we have not surmounted the "barrier," it appears we are climbing in the right direction.

---

[1] Lewis H. Sarett, "Synthetic Organic Chemistry: New Techniques and Targets," presented before Synthetic Organic Chemical Manufacturers Association, June 9, 1964.

[2] E.J. Corey and W. Todd Wipke, "Computer-Assisted Design of Complex Organic Syntheses," Science, 166, 178 (1969).

[3] E.J. Corey, W. Todd Wipke, Richard D. Cramer, III, and W. Jeffrey Howe, "Computer-Assisted Synthetic Analysis: Facile Man-Machine Communication of Chemical Structure by Interactive Computer Graphics," in press. See also adjacent papers by same authors.

[4] E.A. Feigenbaum and J. Feldman, Computers and Thought, McGraw-Hill, New York, 1963, p. 5.

[5] W. Todd Wipke, Thomas M. Dyott, and Joseph G. Verbalis, "Interactive Three-Dimensional Graphical Input for Synthetic Analysis," Abstracts of 161st ACS National Meeting, Los Angeles, Calif. March 28, 1971.

[6] W. Todd Wipke, Thomas M. Dyott, "General Interactive Graphics Language," Princeton Computer Graphics Laboratory, Memo 20,1-1, April 1971.

[7] J.E. Williams, P.J. Stang, and P. von R. Schleyer, Ann. Rev. Phys. Chem., 19, 531 (1968).

[8] Allen Newell, J.C. Shaw, H.A. Simon, "Chess Playing Programs and the Problem of Complexity," in Computers and Thought, E. A. Feigenbaum and J. Feldman, McGraw-Hill, New York, 1963, p. 39.

[9] H. Gelernter, "Realization of a Geometry-Theorem Proving Machine," Ibid, p. 134.

[10] C.G. Feldman, Douglas T. Ross, Jorge E. Rodriguez, "AED-0 Programmer's Guide," Electronic Systems Laboratory, Massachusetts Institute of Technology, Report ESL-R-406, January 1970, p. 56.

[11] The author is indebted to the National Institutes of Health (Grant 1 P07 RR00578-02) and to Princeton University for their support of this work.

Computer-Based Searching of the Current Chemical Literature by the Individual Scientist

Eberhard Kiehlmann
Simon Fraser University, BURNABY 2, B.C., Canada

(Experiences Gained After Two Years of Participation in the CAN/SDI Project, National Science Library, National Research Council of Canada)

At present, some 300,000 papers of chemical interest, published in 12,000 journals, 28 countries and 17 languages, are abstracted annually by Chemical Abstracts. Selective Dissemination of Information (SDI), i.e., a computer-based pre-selection of articles specifically related to the individual scientist's research interest from this enormous number of source documents, appears to be the most efficient way for the active research chemist to keep up-to-date on recent developments in his field.

Since 1965 the titles, keywords, author names and other bibliographic data of all articles covered by the major chemical abstract and title publications--Chemical Abstracts (CA), Chemical Titles (CT) and Science Citation Index (SCI)--have been stored on magnetic tape, and these tapes are available for computer-searching against personal interest profiles prepared by or for individual chemists. The important parameters of publication frequency, journal coverage and time delay between publication of the primary document and release of the tape are summarized in Table I. Since April of this year, the even (physical, analytical, macromolecular and applied chemistry) and odd issues (organic and biochemistry) of CA can be searched separately.

To review briefly the terminology used in writing a "search profile", independent chemical concepts become profile "parameters" which are linked together by AND logic. In the example shown in Figure 1 which attempts to retrieve all current articles on "Hydrogen Bonding in Halo-alcohols" HYDROGEN BONDING, HALOGEN AND ALCOHOL represent three different parameters. Each parameter may consist of several profile "terms" which represent closely related words, phrases or synonyms within the same concept or parameter and which are linked by OR logic. In addition, NOT logic may be used to exclude certain terms. A star at the beginning or end of a term shows the facility of front or end truncation which is used to avoid the listing of many words with the same root. This specific profile will retrieve all papers which contain in their title at least one term of each parameter. The computer printout (Figure 2) lists the words or phrases leading to retrieval, the complete bibliographic reference (mutilated author names, title, journal, page numbers, etc.), abstract number (in case of CAC) and number of the CA or CT issue searched. This printout which is in convenient card form for direct filing is mailed to the chemist weekly (CAC, ISI) or biweekly (CT).

One of the factors contributing to the rather poor acceptance of computer-based search techniques by academic scientists is undoubtedly their belief that the translation of a research topic into computer-readable form is a complicated process which may require the learning of some computer language. Quite to the contrary, profile writing is extremely easy (see Figure 1), and the coding rules can be mastered by any scientist in a matter of an hour. And if he does not wish to do his own coding, nearly every Canadian university library has one or more staff members who have received training as "search editors" and can provide expert assistance. At the present time, the science librarians of the three universities in British Columbia are considering a joint full-time appointment of such a person. All the individual scientist has to provide is a clear narrative description of his search request, a list of 10-20 complete bibliographic references from his file of relevant reprints and perhaps a number of synonyms for certain specialized terms which may be used by other chemists in the field.

The staff of CAS (Chemical Abstracts Service) and NSL (National Science Library) are continually modifying their computer programs in order to simplify coding even further and to add versatility to their search service. Recent improvements include:

a) Elimination of the distinction between title (T) and keyword (K) terms the latter of which were used exclusively for searching the keyword indices of the weekly CA issues. Thus in order to search both titles and index, each term had to be listed twice -- at twice the cost. As a corrolary, front truncation and the use of phrases are now allowed for all search terms.

b) In addition to author names, titles and keywords, type of publication (Books, conference proceedings, dissertations, journal articles, patents, technical reports), language and CA section codes are now searchable.

In order to evaluate the results of searching various data bases against personal interest profiles and to suggest possible profile improvements, NSL mails to each subscriber a "response form" (Figure 3), i.e., a duplicate printout of each retrieved reference which provides three spaces for judging the relevancy of the reference: YES, NO or CANNOT TELL. These response forms, if regularly returned to Ottawa, are analyzed by NSL to measure the profile efficiency in terms of its "precision" which is defined as the ratio of relevant hits (a "hit" is a computer-retrieved reference) to the total number of hits. The five profiles prepared by 11 faculty members of the Chemistry Department of Simon Fraser University (SFU) and searched in the CAN/SDI Project have been found to give an average 25% precision on a total average weekly printout of 20 references per search question, i.e., five relevant papers each week. The so-called "recall" ratio which is obtained by dividing the number of relevant papers retrieved by the total number of relevant papers present on the data base may be used as another quantitative measure of profile efficiency. Although this ratio is hard to measure, regular manual checking of the tables of content of the 25 most important journals publishing in my research area indicates fairly high recall (approximately 65%) although my interests cover the wide field of carbonyl condensation reactions which cannot be easily described in 60 profile terms.

This result shows clearly one of the major advantages of computer based search techniques over manual methods: By simply scanning approximately 20 references arriving on my desk every week, I discover about four out of every six relevant papers published in my research area (assuming 20% precision and 67% recall) and, even more important, this includes articles from some of the most obscure chemical journals. Most of the unretrieved third of relevant papers are generally picked up by checking cited references, manual scanning of tables of content and reading of a few major journals.

Comparing the three major chemical data bases with respect to recall ratio (extent of journal coverage), currency and convenience, I would rate CAC (Chemical Abstracts Condensates) first. Although CA is less current than CT by about 10 weeks, it offers essentially complete coverage of the field and the convenience of listing the abstract number for immediate reference to more detailed information than revealed in the title. Since life science (biology, medicine, etc.) journals are strongly represented on the ISI (Institute for Scientific Information, Philadelphia) tapes, this data base is particularly advantageous for biochemists.

Thus only one question remains to be answered: Can an academic scientist financially afford a subscription to a computer-based alerting service in these times of tight budgets? Subscription to the source tapes is clearly cost-prohibitive for any single university. However, searching these data bases against a large number of personal interest profiles at independent regional information centres like NSL, Ottawa, is financially feasible. According to the new NSL pricing schedule which is based on approximately 700 subscriptions, regular weekly searching of CAC against a 60-term profile will cost $117 annually. At SFU, we have cut the cost to the individual even further by combining the search requests of several people on one profile. However, this technique works only if the interests of the cooperating chemists are divergent enough to avoid relevant computer printouts ending up in one mail box only when they really would have been of interest to two or three participants. Thus with three contributors to one 60-term profile the entire CAC data base can be searched weekly for less than $40 per person per year. Weekly searching of the Science Citation Index for all papers which have cited one specific target reference would cost an additional $3.12 per year. The subscription rates for the ISI Source tape are comparable while CT searching is considerably less expensive.

I am certain that in the near future additional data bases will become available for searching and that wider acceptance of SDI services by academic and industrial chemists will lower the costs even further. The next step I can envisage for the development of these techniques is the searching and printing of entire abstracts instead of bibliographic references alone. The possibility of using computer-based services to keep up-to-date on recent advances in the field of chemical education or, for that matter, in the field of computer-based methods of information retrieval should not be overlooked.

TABLE 1

Tape Service Characteristics of Major Chemical Data Bases

	CAC	CT	ISI
Publication frequency	Weekly	Biweekly	Weekly
Journal coverage	12000	650	2000*
Titles per issue	5800	5000	5-15000
Average time delay	14 weeks	4 weeks	3 weeks
Searchable elements:			
Title	x	x	x
Keyword	x		
Author	x	x	x
Journal	x	x	x
Citation			x

* all fields of science
(approximately 650 chemistry journals)

FIGURE 1

Three-Parameter Search Profile

	Search Terms	AND	AND	but NOT
	HYDROGEN BOND*	*HAL*	ALCOHOL*	PERIOD*
OR:	HYDROGEN BRIDG*	*FLUOR*	*ANOL*	FLUORIMETER*
OR:		*CHLOR*	*ENOL*	THALLIUM
OR:		*BROM*	*CARBINOL*	
OR:		*IOD*	HYDROXY*	

FIGURE 2

Computer Printout of Bibliographic Data

```
HYDROXY, ENOL, CHLOR, HYDROGEN BOND
PERELYGI, I.S., AKHUNOV, T.F.
INFRARED SPECTRA AND HYDROGEN BONDING OF HYDROXYLS
    OF CHLOROSUBSTITUTED PHENOLS .

OPTIKA I SPECTROSKOPIYA
VOLUME 0029, ISSUE 0003, YEAR 1970, PAGE 516-18

PHENOLS**CHLORINATED**IR**HYDROGEN**BONDING**CHLORO**


AN  007977       0803        EN 14       C0274
```

FIGURE 3

Computer Printout Response Form

```
HYDROXY, ENOL, CHLOR, HYDROGEN BOND
PERELYGI, I.S., AKHYNOV, T.F.
INFRARED SPECTRA AND HYDROGEN BONDING OF HYDROXYLS
   OF CHLOROSUBSTITUTED PHENOLS.

OPTIKA I SPECTROSKOPIYA
VOLUME  0029, ISSUE 0003, YEAR 1970, PAGE 516-18

PHENOLS**CHLORINATED**IR**HYDROGEN**BONDING**CHLORO**

IS THIS CITATION USEFUL?  YES   NO   CANNOT TELL

AN  007977       0803        EN  14        C0274
```

Use of the ASTM-IR File for Education

Joseph R. Denk
Curriculum Development Manager
North Carolina Educational Computing Service
P. O. Box 12175
Research Triangle Park, North Carolina 27709

The Dow-ASTM file of coded infrared spectra for 92,000 compounds has become available through several commercial sources[1]. These commercial sources offer a service which includes a running software package to do searches on the file itself. To the knowledge of this author, all of these services are limited to searches in which a infrared spectrum is "identified" from a combination of coded peaks for the unknown spectrum and a limited amount of structural information of the unknown material. The file itself, once available, allows much more versatility in searching and several possibilities for educational innovation.

The American Society for Testing and Materials, ASTM, can supply the original file of 92,000 compounds and an update of 50,000 additional compounds. This data base can be described as having three categories of information for each entry:

(a) Coded spectral data,
(b) Structural Information,
(c) Other chemical information.

Since the file has both inorganic and organic entries, the information in each category differs for these two general chemical classifications.

For organic compounds, the infrared spectral coding is available for bands having over 30 percent absorption in the range of 1.0 to 16.0 microns. Structural information for each entry is summarized in Table 1. Other chemical information includes melting or boiling points and empirical formula data.

---

[1] Among these are Sadtler Corporation, Singer Corporation, and Dow Chemical Corporation.

Inorganic compounds have spectral coding from 11.0 to 35.0 microns for bands over 30 percent absorbance. Structural information is limited to the presence of 92 possible Cations and 75 possible radicals. Again, melting or boiling point information as well as empirical formula data are available.

Besides the obvious usage of such a file for organic qualitative analysis by searching, the large amount of structural information suggested to the author several educational possibilities. An inversion of the searching technique, i.e., going from structural information alone to spectra, would allow semiquantitative studies in the effects of electron-feeding groups, electron-withdrawing groups, steric hindrance, etc., or an active infrared site. Band shifts and complications would be available from a compound "constructed" and searched from the file of published spectra. These possiblities are relevant to chemistry courses covering structural theory, spectroscopy, chemical literature, and undergraduate research besides those courses involving organic or inorganic qualitative analysis.

The author obtained the Dow-ASTM file and constructed a searching system for searching the file with any combination of input information to produce any combination of file information. This system, Infrared Spectral Information System, ISIS, allows organic qualitative analysis through spectral information or through a combination of spectral and structural information. ISIS also allows pulling out spectra that match a combination of structural parameters designed by the user. Further, ISIS allows any combination of these approaches.

TABLE 1

Structural Information for Organic Compounds in the ASTM Infrared File

General	Specific Symbolic Groups
Position(s) of Unsaturation in Chains	Oxygen Groups
Position(s) of Unsaturation in Rings	Nitrogen Groups
Presence of Unsaturation	Oxygen-Nitrogen Groups
Structural Type	Sulfur Groups
Ring Size	Sulfur-Nitrogen Groups
Chain Length	Sulfur-Oxygen Groups
Number of Rings	Sulfur-Oxygen-Nitrogen Groups
Position(s) of Substitution in Chains	
Position(s) of Substitutes in Rings	
Physical State	
Direction of Rotation	
Cis-Trans Isomerism	
Presence of Spiro Arrangement	
Type of Alkyl Group	
Type of Alkenyl Group	
Type of Alkynyl Group	
Presence of Conjugation	
Presence of Heterocyclic Compounds	

Five general statements are involved in ISIS for searching: RUNTYPE, OPTIONS, INCLUDE/EXCLUDE, NOBAND, and SPECTRUM. The RUNTYPE statement allows a search through spectra for identification, a search through structure for spectra, and combinations of these approaches. OPTIONS includes wiggling spectra during searches, subsetting the file, pointing out the coded spectral information on hits, getting the names of the hits (instead of serial numbers), and requesting structural information. INCLUDE/EXCLUDE statement allows the addition of structural information to spectral searches. Spectral searches use the NOBAND and SPECTRUM statements to input spectral information.

During the Spring of 1971, when ISIS first became available, the structure to spectra searching was not implemented and usage was restricted to organic qualitative analysis. Structural theory studies were not attempted except in testing beginning software. ISIS is now almost finished and structural studies will take place in the fall of 1971. To aid in this work, Charles N. Reilly has studied the cleanliness of the file for structural information. These results are yet to be published.

ISIS can do a search for less than $1.00. This factor made the production of another system (for IBM 360 Models 50 and up) worthwhile educationally since commercial searches are ten times more expensive. Information on the availability of the software (searching system in 360 ASSEMBLER and I/O in PL/I) can be obtained from the author.